Libro Vivo de hechizos

LIVIANO

Huayra Linux

Libro Vivo de Hechizos de Huayra Linux – Edición Liviana

Coordinación Equipo Huayra Linux
Lic. Javier Castrillo

Compilación y edición
Lic. Diego Accorinti

Autores/as
Equipo Huayra Linux

Diseño Gráfico e ilustración
Carolina Hortal

Bajo licencia CreativeCommons Atribución-CompartirIgual 4.0 Internacional.

Introducción

Finalmente la fortuna llega a sus manos, porque se encuentra leyendo un libro lleno de secretos, atajos, trucos, y recetas para dominar como nunca antes el **sistema operativo Huayra Linux** y aprovechar al máximo todas sus virtudes.

¿Por qué un libro de hechizos? Porque no hay que dejar de jugar, y si hay que lidiar con lo desconocido, ¡qué mejor que un poco de magia de nuestro lado!

Parte de esa magia consta en que este manual se actualizará automáticamente cuando lo vea en el Escritorio de Huayra, ya que **es un paquete más del sistema** y cuando hagamos modificaciones, llegarán en forma de actualización junto con las demás mejoras de software.

Si llegó a sus manos como un archivo PDF, le contamos que está leyendo la revisión 5. Puede revisar si existe una versión más nueva en este enlace, y también hacernos llegar sus correcciones y sugerencias.

Índice

Introducción....3

Capítulo 1 - Iniciación....7

¿Qué tiene que ver la soberanía nacional con el software?....7

¿Qué es Huayra?....10

¿Por qué usar Huayra?....10

¿Qué aspecto tiene Huayra, me va a resultar complejo usarlo?....10

¿Puedo instalar Huayra en cualquier computadora?....10

¿Huayra es compatible con los formatos de archivos que uso todos los días? 11

¿Puedo usar Huayra aunque no sea parte de una comunidad educativa?....11

¿Es seguro, tengo que usar antivirus?....11

Capítulo 2 - Instalación....12

2.1 Cómo hacer un pendrive live de Huayra....12

2.2 Qué es el BIOS y cómo desactivar Secure Boot....13

2.3 Cómo instalar Huayra en una computadora....17

Capítulo 3 - Actualización....23

3.1 Cómo mantener actualizado a nuestro sistema Huayra....23

3.1.1 Actualizar de forma gráfica....23

3.1.2 Actualizar desde la Terminal de Huayra....25

Capítulo 4 - Sumando hechizos....28

4.1 Cómo instalar aplicaciones extra desde el repositorio....28

4.2 Cómo instalar aplicaciones de terceros....30

4.3 Usando aplicaciones portables....34

Capítulo 5 - Licencias....36

5.1 Si es gratis, ¿es libre?.....36
5.2 Una aplicación en la nube, ¿es libre o privativa?.....36
5.3 Mis estudiantes hicieron una aplicación, ¿Qué licencia le ponemos?.....37
5.4 Mis estudiantes hicieron obras culturales ¿Qué licencia le ponemos?.....37
Capítulo 6 · VideoJuegos.....39
6.1 Juegos libres.....39
6.2 Emuladores.....40
6.2.1 Stella.....41
6.2.2 MAME.....41
6.2.3 mGBA.....41
6.2.4 ScummVM.....41
Capítulo 7 · Magias varias.....42
7.1 Cómo conectarse a internet por WiFi.....42
7.2 Cómo conectar un dispositivo por tecnología Bluetooth.....43
7.3 Soy programador/a ¿Cómo puedo participar en el desarrollo de Huayra?. 46
Capítulo 8 · La Terminal.....47
8.1 “Dame una terminal y moveré el mundo.”.....47
8.2 Comandos.....48
8.3 Comandos de archivos.....50
8.4 resumen de comandos básicos.....51
8.5 Enviar la salida de un comando a un archivo.....53
8.7 El comando grep.....54
8.7 El comando history.....55
8.8 ¿Qué es un “script” ?.....55
Capítulo 9 · Configuración de Audio y video.....58

9.1 Cómo conectar un monitor, televisor o proyector a nuestra netbook.....58
9.2 Configuración de salida de Audio.....62
9.3 Desactivar el sonido de inicio de Huayra.....66
Capítulo 10 · Accesibilidad.....68
10.1 Contraste de colores, agrandar texto y modificar puntero.....69
10.2 Lector de pantalla.....70
10.3 Teclado en pantalla.....71
Apéndice A Obsolescencia programada.....72
¿Qué es la Obsolescencia Programada?.....72

Capítulo 1 - Iniciación

¿Qué tiene que ver la soberanía nacional con el software?

Como individuos que somos, tenemos cada uno de nosotros y nosotras, una construcción diferente sobre determinados conceptos. Muchas de esas construcciones vienen de tiempos de la escuela primaria, donde muchos y muchas de nosotros empezamos a formarnos como ciudadanos. En ese marco, la palabra soberanía probablemente remita a cuestiones puntuales como las Islas Malvinas, a la vuelta de Obligado o la Bandera Nacional. Esta concepción proviene de una historia dinámica, que a partir de la segunda mitad del S XIX encuentra un período de formación de los estados nacionales en medio de una expansión imperialista. Por otro lado, en el mismo período, se construyen las bases de la escuela tal como la conocemos, la que forma los ciudadanos y ciudadanas que el país necesita. Es por eso que la **soberanía**, en tanto poder político supremo de un estado y que lo hace independiente y único, se aprende desde esa **escuela propia, pública y nacional**.

Los tiempos cambian, las tecnologías y metodologías también. Y es por ello que los ataques a la soberanía nacional no se limitan a una cuestión de fronteras o símbolos patrios. Los medios de comunicación, los poderes financieros, los avances de la ciencia y las TIC, proponen otras maneras de atacar la soberanía de las naciones. Un país que no pueda desarrollar sus políticas por estar sojuzgado a las políticas de otro estado, o que dependa tecnológicamente de otro/s, también puede ser que se vea privado de ejercer su legítimo derecho a capacitarse y producir con conocimiento propio; estos pocos ejemplos -reales y actuales- son solo una muestra del ataque que sufren hoy las soberanías nacionales.

En lo estrictamente computacional (por “computacional” entendemos no

solo a la PC / notebook sino a todo aquello que incluya o sea una computadora: smart TV, tablet, smart watch, netbook, consola de juegos, etc.), quienes hayamos ido al menos a una clase de computación sabemos que los dispositivos se componen de *hardware* y *software*. Y también sabemos que es el software el que hace que aquel funcione de una manera determinada. Es el software, en definitiva, el que dirige y ordena las funcionalidades de un dispositivo. Por lo tanto si ese software no es auditable, es decir que no podemos estudiar cómo funciona, estará haciendo cosas que no sabemos a ciencia cierta qué es lo que hacen. Veamos algunos casos que seguramente no nos resultarán extraños:

1. Cuando conversamos con alguien cerca de nuestro teléfono sobre algún tema particular, por ejemplo las remeras de verano, seguramente aparecerán en los días sucesivos avisos en nuestras redes sociales sobre venta de remeras y las ofertas de verano en indumentaria. Esto, claramente, no es casualidad. El teléfono está “escuchando” y catalogando todo lo que decimos para luego enviarnos avisos relevantes para nuestras costumbres. Evidentemente ese proceso de “escucha” de nuestras conversaciones no fue voluntario. Nosotros solo teníamos el celular arriba de la mesa. Bien, el software de ese móvil hizo lo que su fabricante quiso que hiciera: espiarnos.

2. Nuestra computadora, comprada hace un par de años, ya no es lo que era. Los programas tardan una eternidad en abrir cada aplicación, se queda “pensando” continuamente y los “expertos” nos dicen que por más que la instalemos de cero (cosa que hasta ahora había hecho que estuviera rápida nuevamente) ya está vieja y tenemos que comprar otra. Los softwares de ahora no son los de hace dos o tres años y necesitan de una máquina que tenga el doble de RAM que la que tenemos... Esto se llama **obsolescencia programada** y está perfectamente calculado. Nuestras aplicaciones harán exactamente lo mismo que lo que vienen haciendo pero misteriosamente necesitan muchos más recursos que los que necesitan ahora, obligando a nuestros castigados bolsillos a comprar periódicamente nuevos dispositivos. Es insólito que una computadora de tres o cuatro años atrás se considere “obsoleta” solamente porque el fabricante de software hace sus programas pesados y lentos ex profeso.

3. Nos mandan un archivo con una extensión no normalizada (por

ejemplo un pptx, xlsx o docx) y para abrirlo necesitamos de un programa que, o bien lo tenemos que comprar o bajarlo ilegalmente de internet. No hay otra manera de ver ese archivo sino recurriendo a un único programa. Va de suyo que si tenemos una aplicación así en el Estado Nacional, deberíamos comprar siempre al mismo proveedor para mantener activos todos los archivos anteriores ya que si compramos otro software tendríamos los datos en otro formato y no sería compatible con lo actual. Es decir, nos tienen agarrados de los pelos, no podemos buscar una alternativa.

Nuestra computadora, la que usamos todos los días, está llena de trabajos que venimos haciendo desde hace tiempo, tiene guardadas nuestras fotos y archivos importantes, también memoriza sitios webs usuales, aplicaciones más utilizadas… está al día en todo. De repente “nos entra un virus” y toda esa información se pierde, sin más, borrada, ida, inexorable. Nuestra computadora es vulnerable y alguien se aprovechó de esas vulnerabilidades. Y quien nos cobró, de alguna u otra manera, ese software… pues no nos da ninguna solución al respecto.

¿Qué es Huayra?

Huayra GNU/Linux es el primer sistema operativo libre desarrollado por el Estado Nacional. Contiene recursos educativos abiertos y aplicaciones que son útiles tanto en el ámbito escolar, para el cual está diseñado, como para su utilización en cualquier otro contexto: en la PC de tu casa y en tu trabajo, se puede instalar en casi todas las PC de escritorio, notebooks y netbooks. Huayra (como veremos más adelante en el **capítulo 5**) es software libre, y además, es gratuito. Esto significa un cambio de paradigma, ya que su licencia de uso garantiza derechos a los usuarios y las usuarias: tenemos derecho de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software, legalmente.

¿Por qué usar Huayra?

Huayra es una construcción federal en la que vos, tu escuela, tu familia, tus amigos, tu lugar, tu barrio, tu provincia, son protagonistas. Ser parte de un proyecto soberano en el que desde cada lugar de nuestro país podemos participar, donde todos somos iguales. Tu participación desde, por ejemplo, Formosa, puede ser importante para una escuela de Río Grande. O una escuela de Misiones puede trabajar colaborativamente con otra de La Pampa para solucionar un tema en una comunidad de Tucumán. Huayra es colaboración en libertad. Es la herramienta tecnológica, segura y libre que necesita cada joven del siglo XXI de un país que está en plena reconstrucción.

¿Qué aspecto tiene Huayra, me va a resultar complejo usarlo?

Huayra tiene todo lo que ya conocés de un entorno de trabajo basado en escritorio: íconos, ventanas, carpetas, menús con aplicaciones separados temáticamente, tacho para tirar lo que ya no te sirve, etc. A poco de usarlo te vas a sentir “como en casa”, porque el equipo de diseño de Huayra ha pensado cada detalle gráfico para brindarte la mejor experiencia de usuario. Y aunque el diseño amalgama los elementos de una manera armónica, agradable y pensada en el usuario, podés “tunear” tu escritorio usando colores, wallpapers, protectores de pantalla y muchos recursos gráficos que diseñamos y recopilamos para vos. Además tiene programas de diseño y edición para que elabores el material artístico y gráfico que tu imaginación te permita.

¿Puedo instalar Huayra en cualquier computadora?

Sí, podés instalar Huayra en cualquier computadora de escritorio o portátil. Huayra es libre y gratuito. Y si tenés una máquina “viejita” que casi no usás porque te parece lenta…. instalale Huayra y te aseguramos que va a mejorar mucho su rendimiento. También podés correr Huayra desde un pendrive, sin tocar los datos que tengas en tu disco rígido… probá Huayra todo lo que quieras desde allí y cuando te sientas seguro, instalalo definitivamente en

tu computadora.

¿Huayra es compatible con los formatos de archivos que uso todos los días?

Claro que sí. Huayra lee, edita y escribe archivos .jpg, .pdf, .xls, .doc, .png y todos los que usás en tus aplicaciones. Te asegura compatibilidad con todos los documentos y archivos que tengas en la actualidad. También es 100% compatible con casi todo el hardware disponible en nuestro país. No estamos del lado de la obsolescencia programada, sino que queremos que esa impresora que tenés hace 10 años siga funcionando todos los días.

¿Puedo usar Huayra aunque no sea parte de una comunidad educativa?

¡Por supuesto! Más allá de su orientación pedagógica, Huayra está 100% listo para cualquier uso que le quieras dar a tu computadora. Tiene aplicaciones para oficina, manejo de redes sociales, reproductores multimedia, agenda, calendario, avisos, internet full, compatibilidad asegurada con servicios de streaming de películas o música… Huayra es el sistema operativo ideal para el uso diario de tu computadora.

¿Es seguro, tengo que usar antivirus?

En el software libre no usamos antivirus tal como los conocés. Como el código fuente está disponible para leerlo, aprender y modificarlo; no es posible “ocultar” un virus o un programa malicioso. Todas las aplicaciones que están incluidas en Huayra pasan por tres instancias de testeo a los efectos de que uses cada una de ellas con seguridad, tranquilidad y eficiencia. Despreocupate de pesados antivirus y todas esas cosas del pasado. Que tu tiempo de uso sea para divertirte, trabajar o estudiar. Solo ocupate de actualizar el sistema cuando te lo pida, dale “aceptar” y relajate. Ni siquiera tendrás que dejar de usar la computadora mientras se actualiza.

Capítulo 2 - Instalación

2.1 Cómo hacer un pendrive live de Huayra

Para empezar debemos descargar la ISO de **Huayra** desde la sección de descargas. También necesitamos un pendrive de al menos **8 Gb**.

ATENCIÓN: La información del pen-drive se borrará por completo, por lo tanto no debe contener nada importante.

Una vez más: **Todo el contenido del pen-drive será eliminado**.

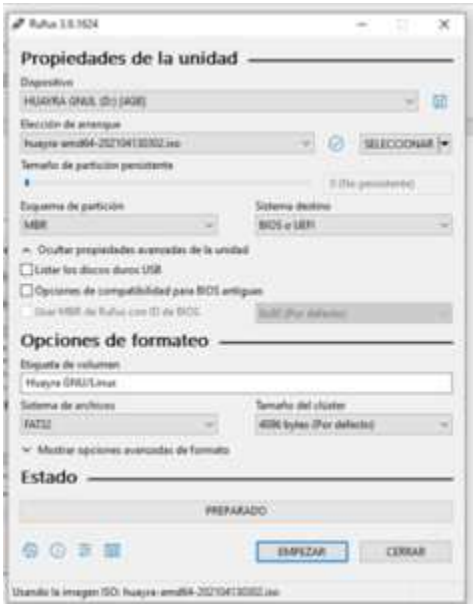
2.1.1 Si ya tenés Huayra

En Accesorios vas a encontrar “Grabador de imágenes USB”. Solo tenés que elegir el archivo .iso que descargaste y el dispositivo USB destino, y pulsar “**grabar**“.
Una vez que tengas listo el pen-drive podrás comenzar con la instalación.

2.1.2 Desde otro sistema operativo

Vamos a necesitar descargar una aplicación llamada “RUFUS” desde acá. Una versión de RUFUS también se encuentra incluída dentro del archivo iso de Huayra, en la carpeta “install”.

a) Abrimos la aplicación RUFUS.
b) Elegimos el dispositivo USB.
c) Seleccionamos la imagen que descargamos y hacemos clic en empezar.



Esperamos unos minutos hasta que se complete la barra de estado, cerramos Rufus y ¡listo! Ya podés usar tu **pendrive Huayra/GNU Linux**.

Una vez que tengas listo el pen-drive, podrás comenzar con la instalación.

2.2 Qué es el BIOS y cómo desactivar Secure Boot

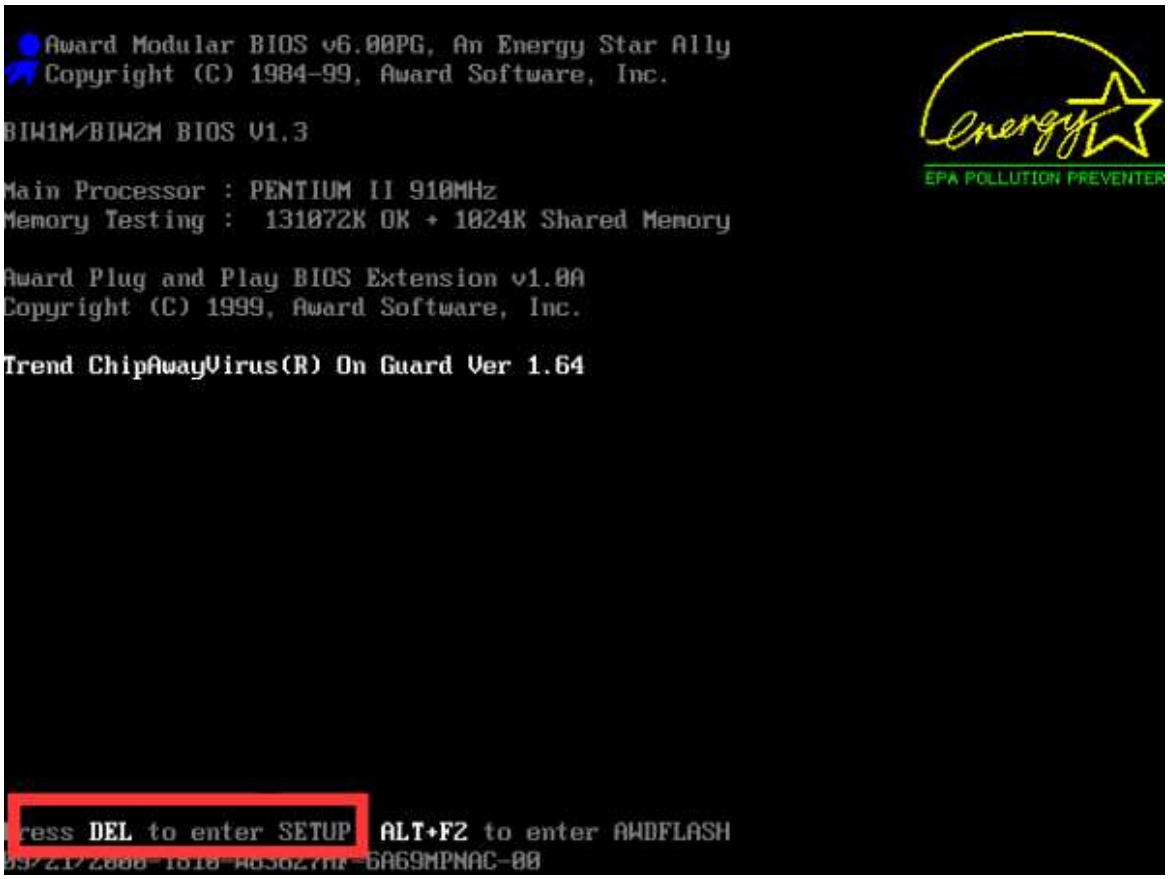
El BIOS, del inglés “*Basic Input Output System*” o “Sistema básico de entrada / salida” es un programa que se encuentra en la placa madre y se encarga de configurar y administrar aspectos relacionados al *hardware*, la fecha, los dispositivos de almacenamiento y el orden de inicio o *booteo*, entre otras cosas.

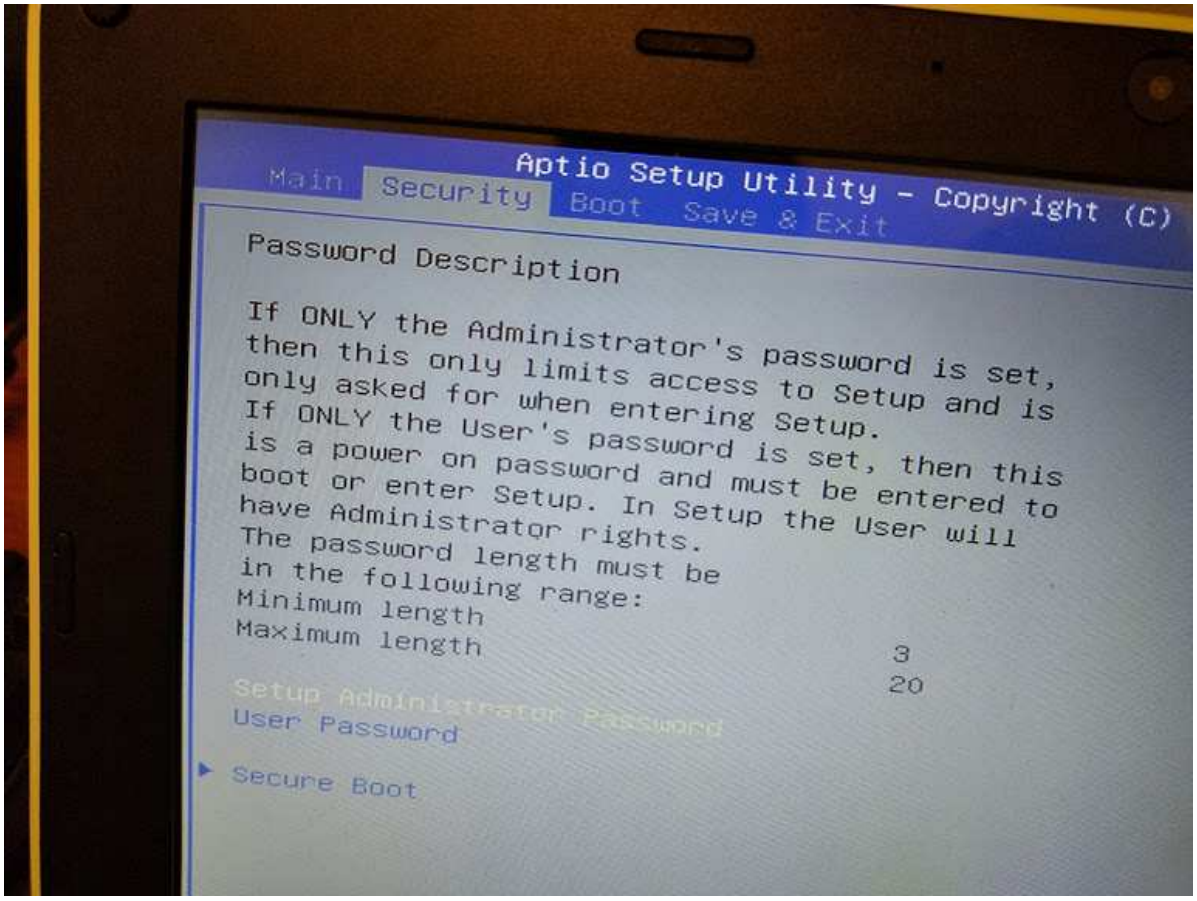
Para acceder a este programa, apenas encendemos la computadora y antes de que comience a cargarse el sistema operativo, debemos presionar una tecla. Dicha tecla cambia en cada marca y modelo de computadora, algunas teclas comúnmente usadas para acceder al BIOS son: F1, F2, F7, F8, F10, F11, F12, ESC, DEL. Por suerte la mayoría de las máquinas muestran un mensaje en la primera pantalla que anuncia cuál es la tecla para acceder, similar a:

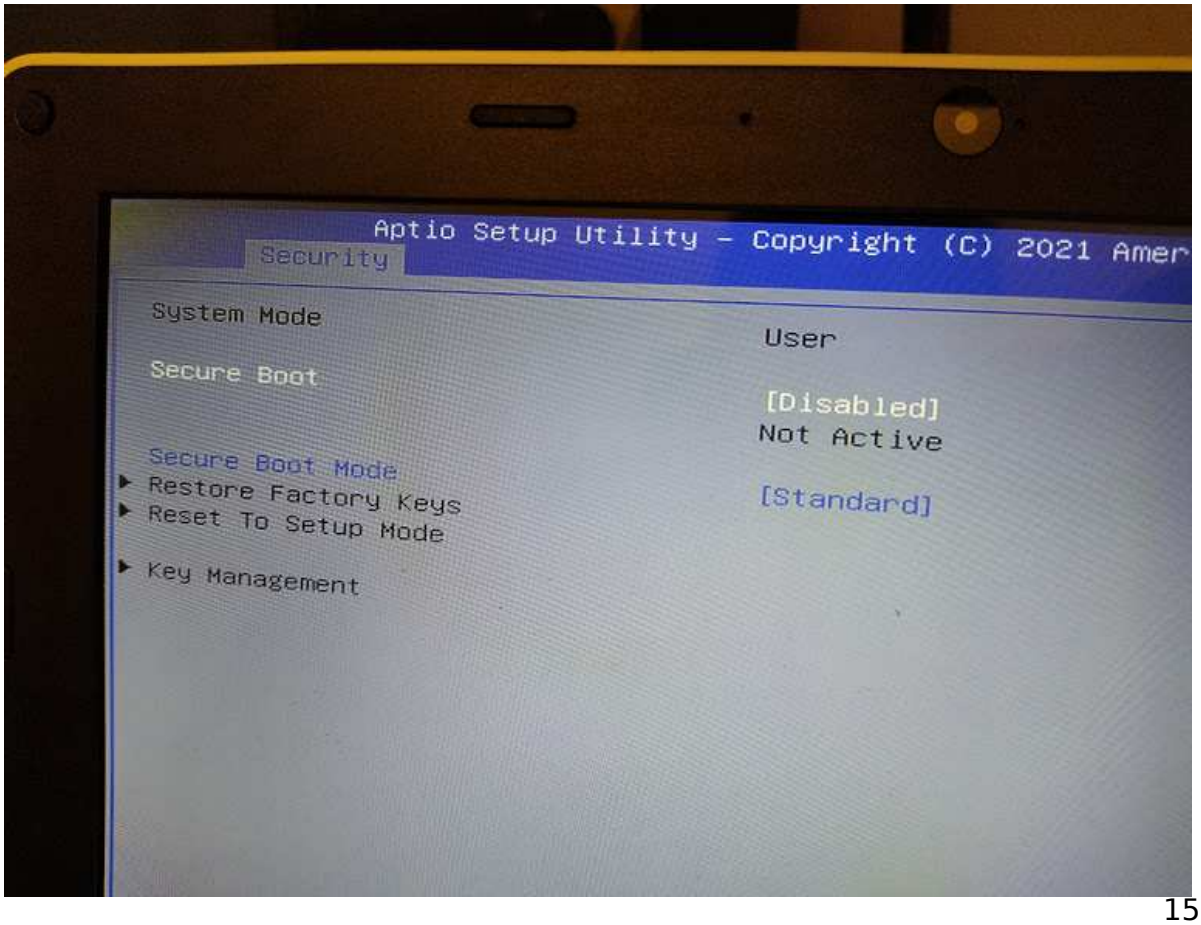
"**Presione [tecla] para acceder a BIOS.**"

"**Press [tecla] to enter SETUP.**"

En la siguiente imagen de ejemplo, observamos que la tecla para ingresar al SETUP / BIOS es "DEL".




Aptio Setup Utility – Copyright (C)
Main Security Boot Save & Exit
Password Description
If ONLY the Administrator's password is set, then this only limits access to Setup and is only asked for when entering Setup.
If ONLY the User's password is set, then this is a power on password and must be entered to boot or enter Setup. In Setup the User will have Administrator rights.
The password length must be in the following range:
Minimum length 3
Maximum length 20
Setup Administrator Password
User Password
Secure Boot


Aptio Setup Utility – Copyright (C) 2021 Amer
Security
System Mode User
Secure Boot [Disabled]
Not Active
Secure Boot Mode [Standard]
Restore Factory Keys
Reset To Setup Mode
Key Management

Cuando vamos a instalar Huayra en una computadora, es necesario que la opción "*Secure Boot*" esté desactivada. Para ello entramos al BIOS y buscamos en la sección de "seguridad" o "boot" hasta encontrar "*Secure boot*" y lo desactivamos.

Una vez desactivado, guardamos los cambios y salimos.

Importante
Necesitamos GUARDAR LOS CAMBIOS antes de salir.

2.3 Cómo instalar Huayra en una computadora

El sistema operativo **Huayra Linux** puede ser instalado en computadoras de escritorio (desktop PC), notebooks y netbooks. Para ello tenemos que iniciar desde el pendrive que creamos en la clase 1, una vez en el Escritorio:

a. Hacé doble clic en el icono "Instalar Huayra GNU/Linux".

NOTA IMPORTANTE: Algunas máquinas montan automáticamente el disco rígido, si en el escritorio aparece un ícono de una unidad de disco como el que se ve en la siguiente imagen marcado con un rectángulo verde (la cantidad de GB puede variar), **hace falta "desmontar" la unidad** para una correcta instalación.

Para desmontar la unidad tenés que hacer clic derecho sobre el ícono -> desmontar. Listo, ahora si podés continuar con la instalación haciendo clic en "Instalar Huayra GNU/Linux".

b. Cuando inicia el instalador elegí el idioma y luego **Siguiente**.

c. Seleccioná tu ubicación y luego “**Siguiente**”.

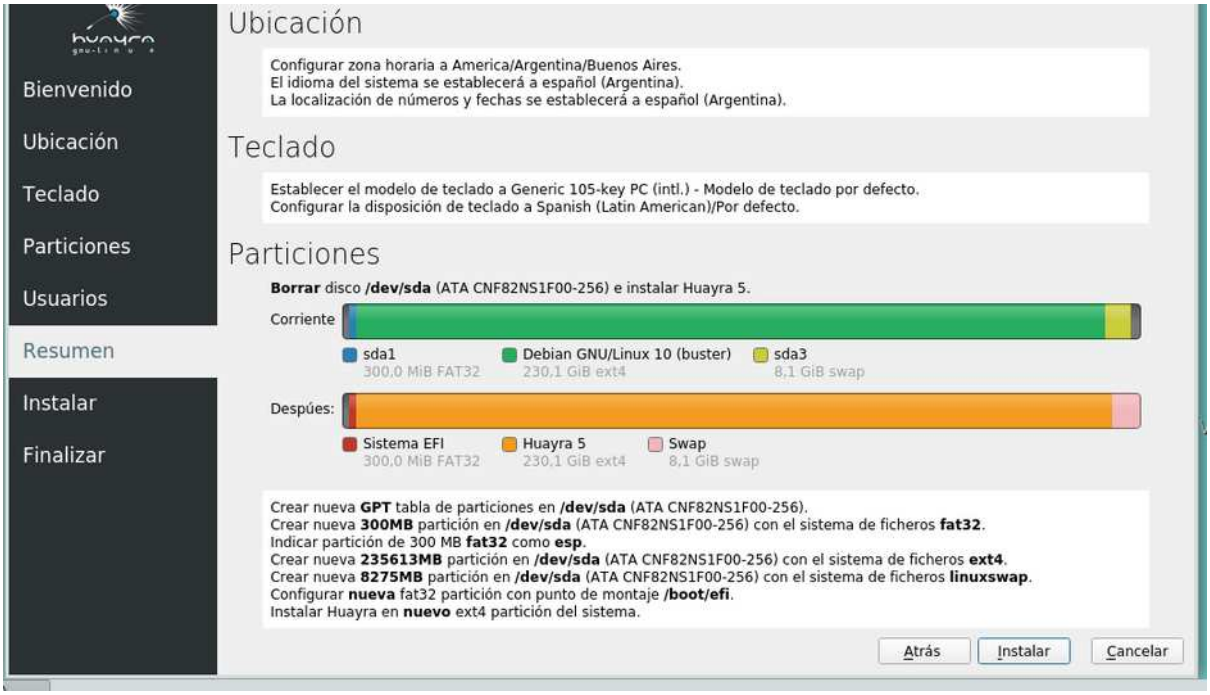
d. En la configuración de teclado seleccioná “Spanish (Latin American)”.

e. Ahora debemos elegir dónde instalaremos Huayra.

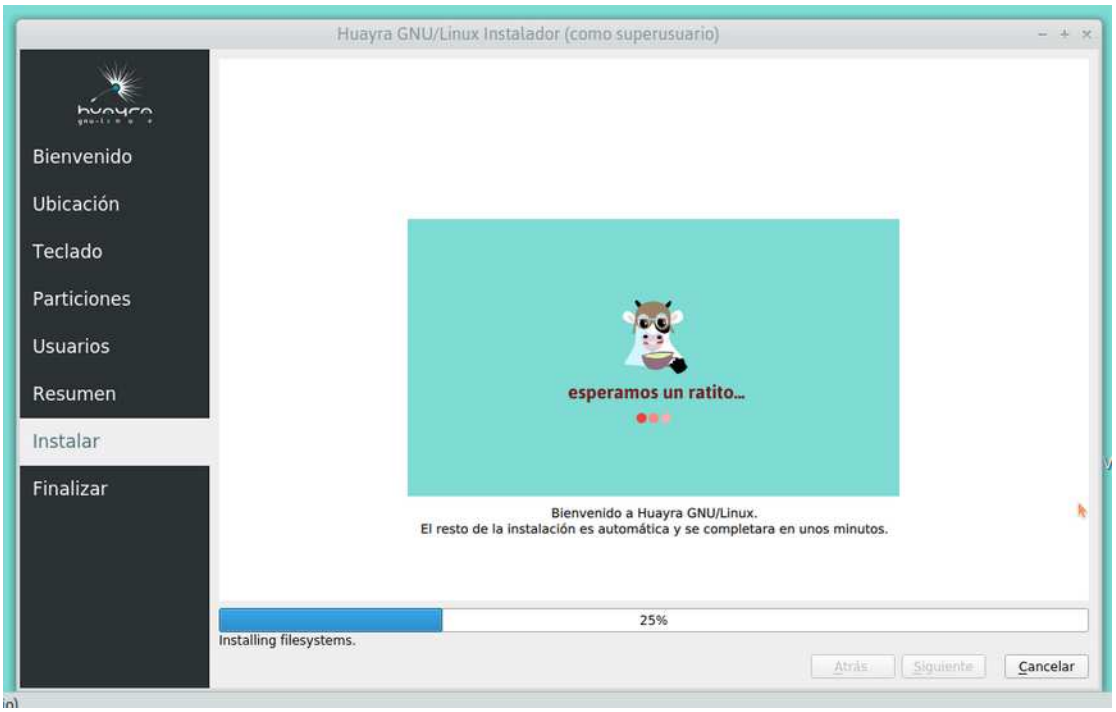
Podemos reemplazar una partición anterior de Linux, instalar junto a otro sistema operativo o elegir la opción: “Borrar disco” (recomendada). Con esta opción el instalador va a crear todas las particiones necesarias para crear una instalación “limpia”.

f. Creá tu usuario y elegí una contraseña.

En “Resumen” aparecen todas las opciones que elegiste. Si todo está correcto presioná “**Instalar**”.



g. Comenzará el proceso de instalación automáticamente.



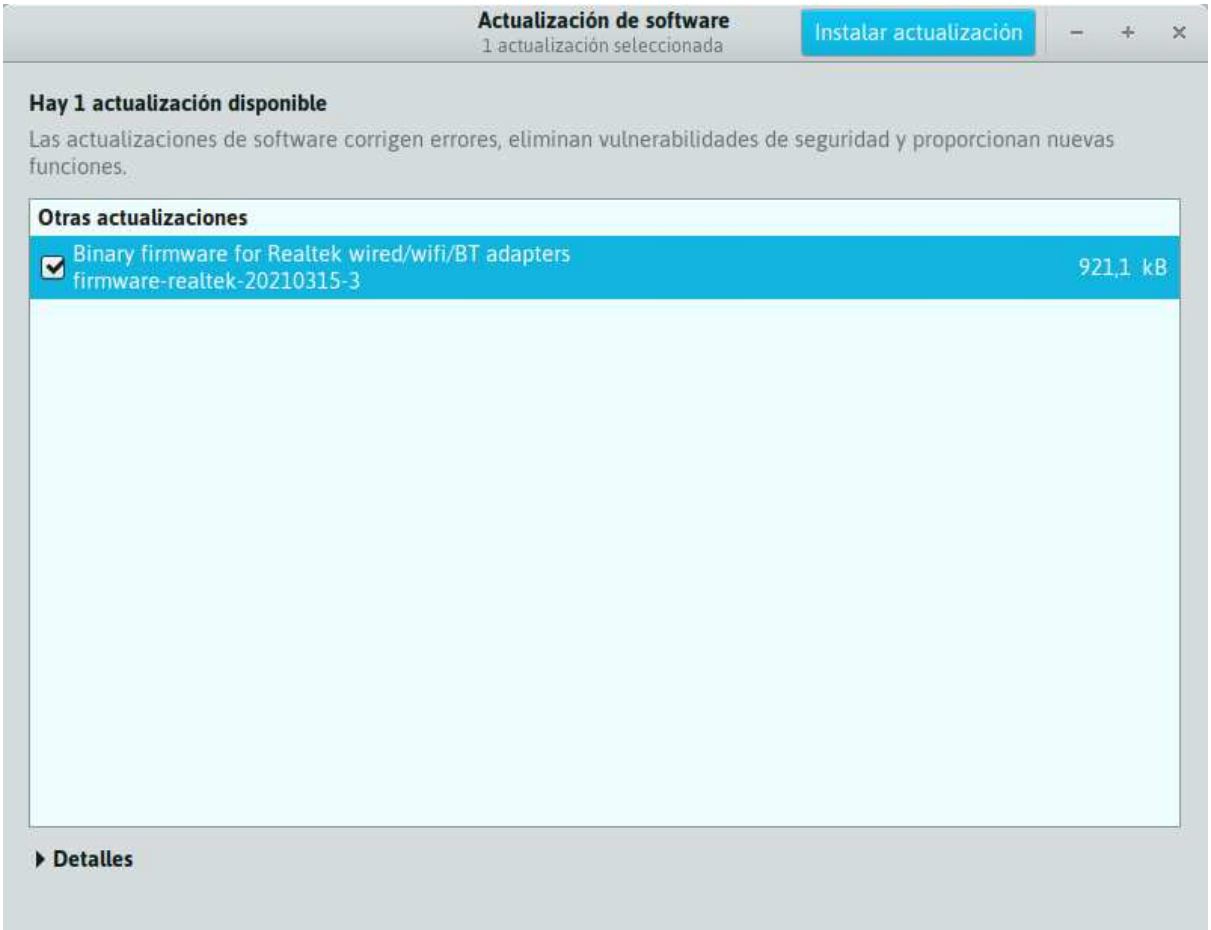
h. Después de unos minutos, aparecerá un mensaje que te avisará que Huayra se ha instalado con éxito. **¡Listo!** Tu equipo ya tiene **Huayra GNU/Linux**.

Capítulo 3 - Actualización

3.1 Cómo mantener actualizado a nuestro sistema Huayra

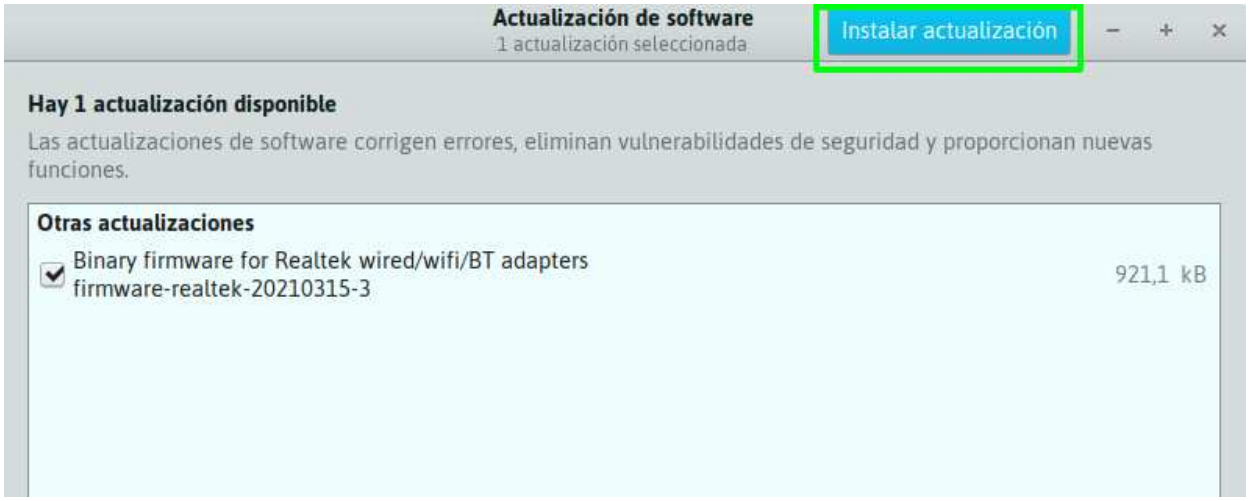
Hay dos formas de actualizar Huayra, una de ellas es con una aplicación en forma gráfica y la otra es mediante la Terminal de Huayra.

3.1.1 Actualizar de forma gráfica



Vamos a utilizar una aplicación que encontramos en el menú: **Aplicaciones** -> **Herramientas del sistema** -> **Actualización de software**.

En la ventana aparecerá el detalle de los paquetes que tienen actualizaciones, solo tenemos que presionar el botón que se encuentra al lado del título de la ventana "**Instalar Actualización/es**".



3.1.2 Actualizar desde la Terminal de Huayra.

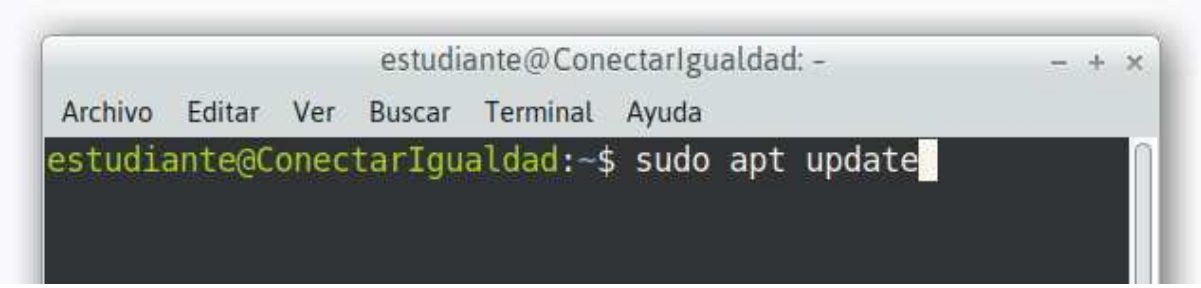
La Terminal de Huayra (también conocida como "consola") es una herramienta que nos permite realizar infinidad de tareas sin necesidad de ventanas o botones. En este caso vamos a ver cuales son los comandos necesarios para actualizar el sistema:

Abrimos una Terminal desde el menú **Aplicaciones** -> **Herramientas del Sistema** -> **Terminal de Huayra**. Si queremos hacerlo con más celeridad, podemos presionar las teclas "**CONTROL**" + "**ALT**" + "**T**" y listo.

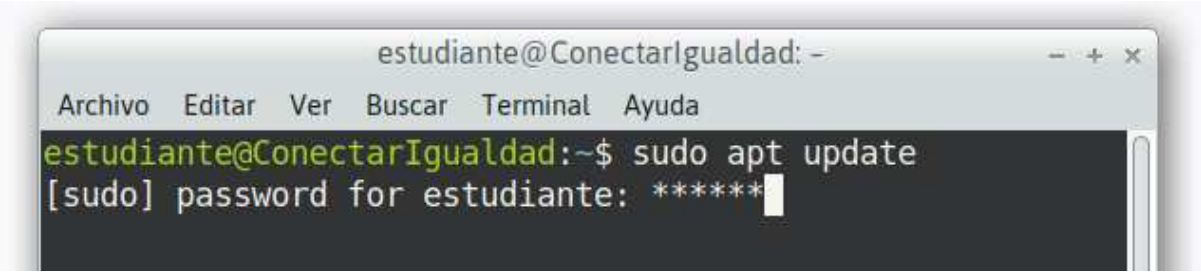
Ahora escribimos el comando para actualizar la información de los repositorios:

$ **sudo apt update**

Y presionamos [**ENTER**].



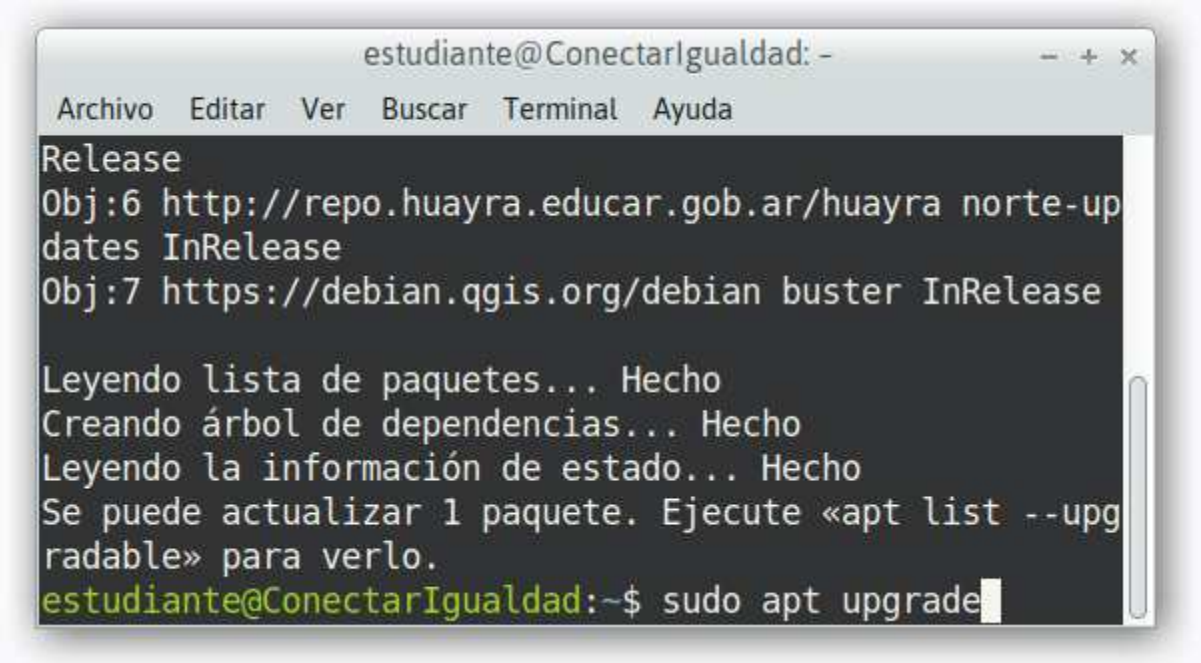
Nos preguntará la contraseña, la ingresamos y presionamos [**ENTER**].



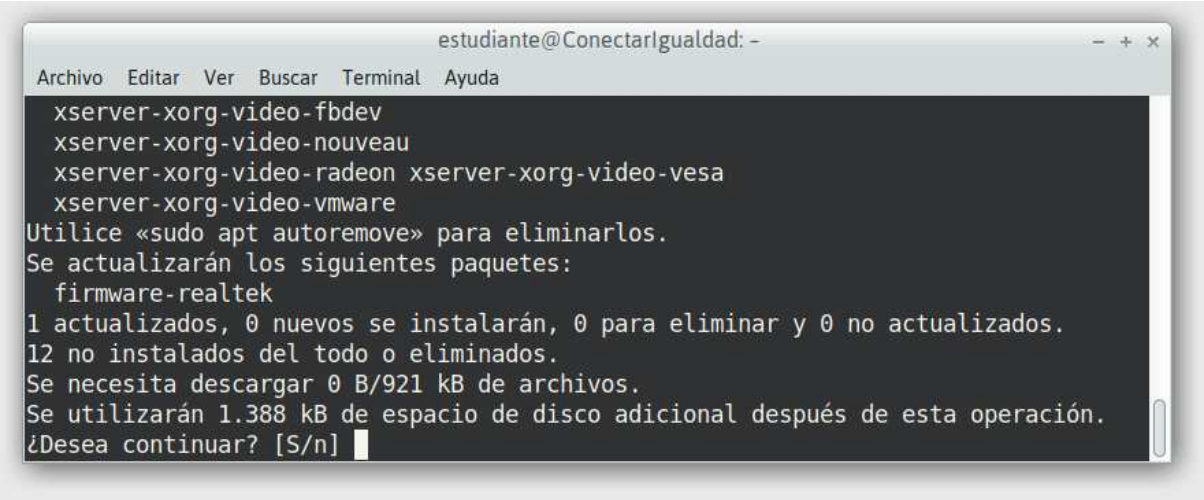
Un vez que haya terminado el proceso, ingresamos el comando para actualizar:

$ **sudo apt upgrade**

Y presionamos [**ENTER**].



Se nos informará los paquetes pendientes a actualizar, así como la cantidad de espacio en disco necesarios, seguidos de una pregunta de confirmación "***¿Desea continuar? (S/n)***" a la que tenemos que responder "**S**" o simplemente [**ENTER**] ya que S es la opción por omisión.



Esperamos a que finalice el proceso y ya tenemos nuestro Huayra **completamente actualizado**.

Capítulo 4 - Sumando hechizos

4.1 Cómo instalar aplicaciones extra desde el repositorio

Las aplicaciones en Huayra se organizan en lo que se llama «paquetes». Instalar programas nuevos es una de las acciones que podemos realizar. El sistema operativo Huayra trae por defecto más de 2500 paquetes instalados y hay más de 57000 disponibles para instalar. Todos están probados y pueden usarse de manera segura. Podemos instalar cualquiera de ellos desde los repositorios oficiales de la siguiente manera:

1. Para comenzar abrimos la aplicación "**Software**", la encontramos en el menú: **Aplicaciones** -> **Herramientas del Sistema** -> **Software** . Dicha aplicación nos permite añadir o quitar software.

2. A la izquierda tenemos una lista de categorías con todas las aplicaciones disponibles para instalar. Si ya sabemos el nombre, podemos buscarla directamente en la barra superior.

3. Para este ejemplo vamos a suponer que no lo sabemos, entonces buscaremos la aplicación por categoría. Elegimos una, en este caso y a modo de ejemplo seleccionamos "Juegos".

4. Ahora simplemente elegimos de la lista (tildándola) la aplicación que queremos (¡pueden ser varias!). Una vez que terminamos de elegir, presionamos el botón azul "Aplicar cambios" (arriba a la derecha). A modo de ejemplo, seleccionamos el juego "Extreme Tux Racer".

5. Nos pedirá la contraseña de Administrador (necesaria para efectuar cambios en el sistema). La clave que viene preconfigurada en Huayra es «estudiante» (sin las comillas, todo en minúsculas). Luego presionamos el botón "**Autenticar**".

¡Listo, en menos de un minuto la aplicación se ha instalado y está disponible en el

menú! La encontramos en **Aplicaciones** > **Juegos** > **Extreme Tux Racer**.

4.2 Cómo instalar aplicaciones de terceros

Puede suceder que la aplicación que necesitamos no se encuentre en los repositorios oficiales (Debian + Huayra) pero el desarrollador ofrece un paquete del software, esto sería un archivo con la extensión ".deb" (dicha extensión hace referencia al sistema de empaquetado de Debian, utilizado por Huayra Linux).

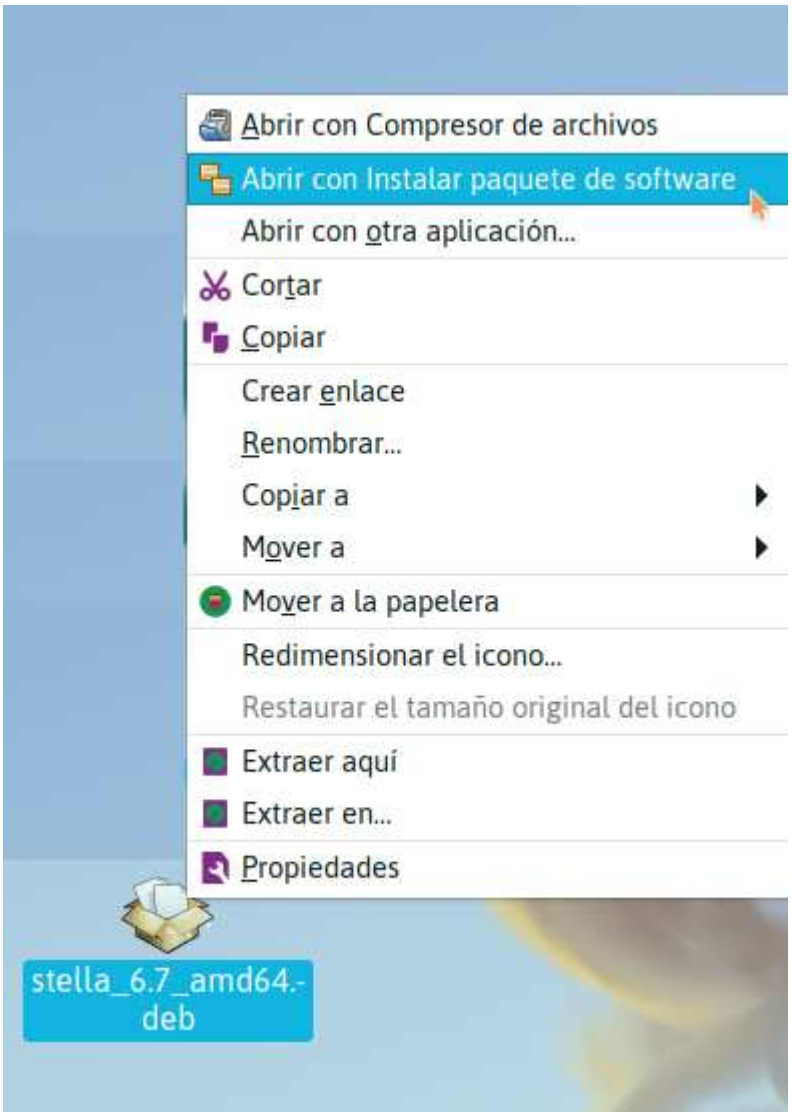
Para hacer un ejemplo, vamos a suponer que el emulador de Atari 2600 llamado "Stella" no estuviera en los repositorios. ¿Cómo haríamos para instalarlo en Huayra?

Los pasos a seguir en este caso son:

a) Descargar el archivo ".deb" desde la página del desarrollador: https://github.com/stella-emu/stella/releases

b) Vamos a ver una lista con el mismo software de muchas "formas" distintas, según la arquitectura y el sistema operativo destino. La que nos interesa es la que tiene extensión ".deb". La mostramos a continuación en la imagen. En cada caso será distinto, pero siempre debemos buscar el ".deb" si es que existe.

c) Una vez descargado el archivo, hacemos clic derecho sobre el mismo y seleccionamos la opción "**abrir con Instalar paquete de Software**".



Nos preguntará si queremos instalar el paquete. Presionamos “**Instalar**”.

Como siempre,nos preguntará por la contraseña de autenticación, la ingresamos y ¡listo! Ya tenemos instalada la aplicación.

Nota: A veces algunos programas necesitan de ciertas "dependencias", es decir, dependen de que otros paquetes estén instalados para poder funcionar.

Cuando instalamos aplicaciones por vía repositorio, estas dependencias se resuelven automáticamente. Pero cuando instalamos por nuestra cuenta, deberemos resolverlas también por nuestra cuenta. ¿y cómo se hace? Vamos a verlo con un ejemplo:

Queremos instalar la aplicación A. Al intentarlo, nos sale un error de que hace falta el paquete "B" para que "A" funcione.

Ok, instalamos entonces B (podemos buscarlo en los repos, quizás este y nos facilita mucho la tarea). Una vez instalado B, volvemos a instalar A, esta vez se instalará correctamente.

El peor escenario será que hagan falta muchas dependencias, o que éstas tengan a su vez otras dependencias incumplidas.

Por este motivo recomendamos, siempre que sea posible, instalar desde repositorios.

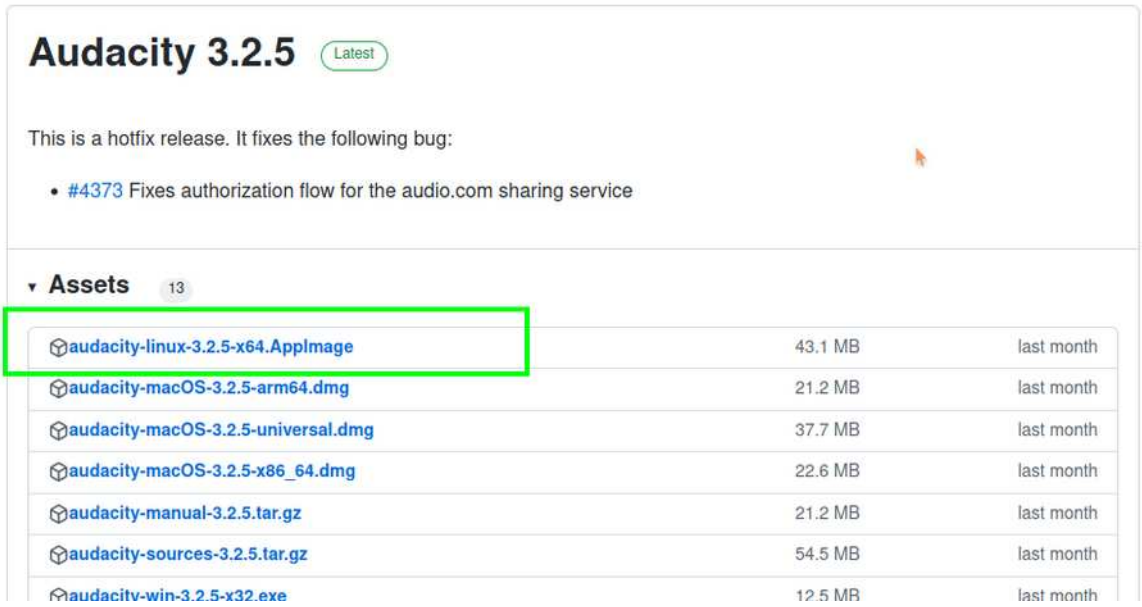
4.3 Usando aplicaciones portables

También existen una serie de formatos “portables” para distribuir aplicaciones. ¿De qué se trata? Bueno, en Linux las aplicaciones están divididas en paquetes porque muchos de ellos son comunes a más de una aplicación, entonces con este sistema modular ahorramos espacio, ganamos velocidad y hacemos que todo sea más “sustentable” ya que las partes de código de las aplicaciones se pueden reutilizar.

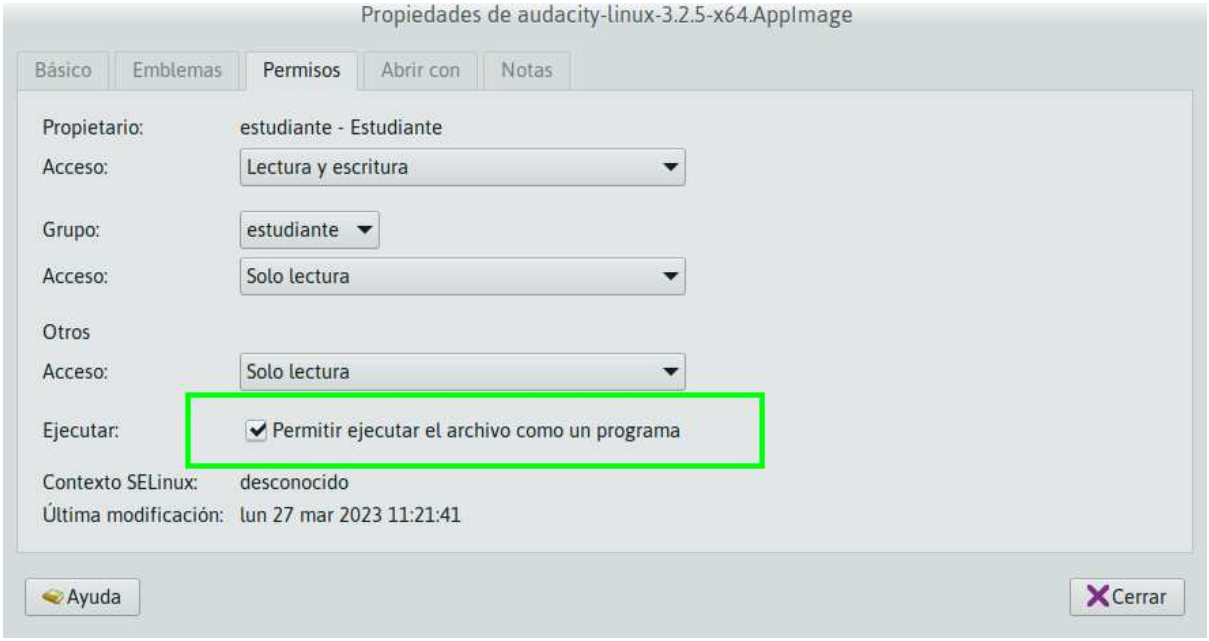
Sin embargo existe otra forma (muy utilizada en el software privativo como forma de oscurantismo) que es meter todo lo necesario para que la aplicación funcione dentro de la misma aplicación. No importa si repetimos librerías, ni si desperdiciamos espacio, el objetivo es no depender de nada y lograr correr la aplicación.) Este tipo de empaquetado portable tiene la ventaja, justamente, de no depender de versiones específicas de paquetes (conocidos como librerías) por lo que funcionan sin más.

Dos de estos sistemas son **Flatpak** y **AppImage.**

Por ejemplo, si queremos utilizar la última versión del editor de audio Audacity, vamos a su **repositorio oficial** y en la sección de descargas vamos a encontrar un archivo con extensión “.AppImage”.



Lo descargamos. Antes de poder ejecutar la aplicación hace falta un paso más. Buscamos el archivo descargado y hacemos **clic derecho sobre el archivo** -> **Propiedades** -> **Permisos** y tildamos la opción **“Permitir ejecutar el archivo como un programa”**, como mostramos a continuación:



Finalmente, haciendo doble-clic sobre el archivo ejecutamos la aplicación.

Capítulo 5 - Licencias

5.1 Si es gratis, ¿es libre?

En el idioma inglés, “gratis” y “libre” se dicen de la misma forma: free. Por ello, para aclarar la ambigüedad, hace falta especificar que “free as freedom, not as free beer” (libre de libertad, no de cerveza gratis). Por suerte nuestro idioma es muy rico y tiene un término para cada caso.

Está claro que algo es gratis si no hace falta pagar dinero para obtenerlo. Sin embargo, a pesar de que mucho software es gratis (por ejemplo, aplicaciones de celulares) no implica que terminemos pagando su costo de otras formas, por ejemplo: publicidades molestas constantes, limitación de uso por determinado tiempo, funcionalidades desactivadas que se activan al pagar, aceptación de términos y condiciones con los que cedemos datos personales al mejor postor, sin saberlo.

Libre es algo completamente distinto, se refiere a que la **licencia de este software** nos **garantiza derechos a los usuarios y las usuarias**: tenemos derecho de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software, legalmente. Para que un software sea libre debe cumplir con dos condiciones, tener una licencia que lo especifique y debemos tener acceso al código fuente del mismo. (condición necesaria que permite estudiarlo, auditarlo y modificarlo).

Huayra Linux es **software libre**, y además es **gratuito**.

5.2 Una aplicación en la nube, ¿es libre o privativa?

Para comprender rápidamente cuál es este tipo de software vamos a mencionar algunos a modo de ejemplo: Google Drive, Microsoft Office 365, Prezi,

Trello, Zoom, etc. Este tipo de aplicaciones se denominan “en la nube” o “SAAS” (software as a service, software como servicio) no son libres, ya que solo podemos usarlas, por un tiempo determinado, bajo la supervisión de quien las brinda, entregando datos personales y/o corporativos, y no podemos conocer cómo están hechas, ni modificarlas. O sea, no cumplen con ninguna de las libertades y derechos que mencionamos anteriormente y que son los que otorga el software libre a sus usuarios y usuarias. Por lo tanto, decimos que el SAAS / aplicaciones en la nube califican como “privativas” o no-libres.

5.3 Mis estudiantes hicieron una aplicación, ¿Qué licencia le ponemos?

Huayra trae incorporadas varias herramientas para programar, las podemos encontrar en el **menú principal** -> **Aplicaciones** -> **Programación**. El software libre es ideal para ello ya que permite mirar dentro del código, ver cómo está hecho un software e incluso ser modificado. Y cuando estamos aprendiendo a programar no hay cosa mejor que volcar los conocimientos en un proyecto de desarrollo, que muchas veces escala y se convierte en una aplicación funcional.

¿Qué mejor que ese proyecto sea publicado **también** como **software libre**? Para que pueda ser aprovechado, mejorado y compartido por toda la comunidad.

Para ello debemos:

1) Agregar al proyecto un archivo LICENCIA.txt que especifique que el software es distribuido bajo la licencia GPL 3.0 (ver más info acá) o cualquier otra licencia libre.

2) Subir todos los archivos del proyecto a un repositorio público, por ejemplo en Github o GitLab.

3) Publicar el repositorio en redes sociales vinculadas a la educación y al software libre. Por ejemplo, en las de Huayra Linux.

5.4 Mis estudiantes hicieron obras culturales ¿Qué licencia le ponemos?

Del mismo modo, cuando trabajamos en proyectos cuyo resultado es una

obra cultural, por dar un ejemplo, un collage sobre algún tema histórico, podemos publicarlo con una licencia que permita la reutilización de esa obra para hacer obras derivadas, reversionarla, mezclarla, es decir, reutilizarla como insumo para generar nuevas obras.

Las licencias más indicadas para ello se llaman Creative Commons, y existen distintos niveles de libertad que podemos otorgar o restringir.

¿Las licencias Creative Commons son válidas en Argentina?

Sí. Es el ejercicio pleno del derecho de autor tal como lo establece el **art. 2 de la ley argentina 11.723**: *"el derecho de propiedad de una obra científica, literaria o artística, comprende para su autor la facultad de disponer de ella, de publicarla, de ejecutarla, de representarla, y exponerla en público, de enajenarla, de traducirla, de adaptarla o de autorizar su traducción y de reproducirla en cualquier forma"*.

Las licencias Creative Commons son **una autorización por adelantado por parte del autor** para ciertos usos sobre su obra. La ley dispone que es el autor quien debe autorizar cualquier actividad relacionada con la obra, pero no especifica en qué momento debe dar la autorización, por lo cual la autorización previa es tan válida como la autorización dada al momento de consultarlo.

Más información sobre licencias culturales libres en https://creativecommons.org.ar.

Capítulo 6 · VideoJuegos

IMPORTANTE: En este capítulo vamos a mostrar juegos libres pero también varias opciones de juegos que son no-libres/privativos; estamos a favor claramente del software libre y quisiéramos que haya una oferta mucho mayor y eso puede suceder en adelante si el mecanismo se pone en marcha: cuanta más gente use y programe software libre, más juegos libres serán publicados. Pero es importante saber que no es necesario usar un sistema operativo privativo para acceder a esos juegos privativos, muchos de ellos podemos jugarlos en Huayra. Además, al final del capítulo mostramos Godot Engine, que es un motor completo y libre de desarrollo de videojuegos 2D y 3D incluido en nuestro sistema operativo.

6.1 Juegos libres

Las posibilidades a la hora de jugar en Huayra son muchas, podemos comenzar buscando en la Tienda de Software de Huayra la categoría "**Juegos**", en donde encontraremos gran cantidad de juegos de software libre disponibles para su instalación.

Aplicaciones -> **Herramientas del sistema** -> **Tienda de software de Huayra**

6.2 Emuladores

Si te apasiona la vieja escuela de los videojuegos, en Huayra tenés disponibles muchísimos emuladores para instalar que te permiten probar juegos de otra época. Muchas de las consolas emuladas funcionan con cartuchos, se conoce como “ROM” al volcado de la memoria interna de un cartucho físico a un archivo digital, para poder ejecutar el juego en un emulador. Huayra no incluye ninguna ROM, pero se pueden encontrar fácilmente en muchos sitios de Internet.

6.2.1 Stella

Con Stella podemos emular la famosa consola **Atari 2600 VCS**, algunos de sus juegos legendarios son: E.T, Space Invaders, River Raid, Enduro, Berzerk y Adventure.

6.2.2 MAME

MAME es un emulador de máquinas recreativas de salón, más conocidas como “Arcades”. No podés perderte: Snow Bros, Bubble Bobble, Golden Axe, Tetris, Street Fighter 2 y muchos más.

6.2.3 mGBA

mGBA es un emulador de la consola Game Boy Advance, donde podrás jugar juegos como Metroid Fusion y Fire Emblem.

6.2.4 ScummVM

Esta aplicación te permite jugar un tipo de juego característico de los 90 's que tienen mucho que ver con la literatura: las aventuras gráficas. Entre los clásicos del género están: The secret of Monkey Island, Day of the Tentacle, Loom, The Dig y Grim Fandango.

Capítulo 7 · Magias varias

7.1 Cómo conectarse a internet por WiFi

1. Vamos a buscar en la barra de menús superior derecha (también llamada Panel) el ícono de red, que ya vimos.

2. Al hacer clic sobre él se despliega el menú contextual -que lista las conexiones de redes disponibles-, donde seleccionaremos nuestra red inalámbrica wifi (que identificamos por su nombre). Si es la primera vez que nos conectamos, **nos va a pedir la contraseña.**

3. Una vez que logramos establecer la conexión, nos muestra un cartel de

confirmación.

4. Vemos que en la barra superior del lado derecho cambia el ícono por el de red conectada, que muestra mediante líneas verdes en forma de arcos la intensidad de la señal (cuantas más líneas, mayor intensidad de señal de red).

7.2 Cómo conectar un dispositivo por tecnología Bluetooth

1.En esta ocasión vamos a conectar un mouse inalámbrico Bluetooth (también hay mouse inalámbrico por radiofrecuencia, pero en ese caso se ocupa uno de los puertos USB, y no hay que setear la conexión por software). En la barra de menús superior derecha, hacemos clic con el botón derecho del mouse sobre el ícono de Bluetooth y, al desplegarse el menú contextual, seleccionamos **Configurar un dispositivo nuevo.**

Los tres puntos al final nos indican que se abrirá un cuadro de diálogo para configurar la acción (es un criterio general, cuando un menú tiene los tres puntos ocurre eso).

2. Al abrirse el asistente de configuración, seleccionamos **Siguiente**

3. Cuando pasamos a la otra pantalla seleccionamos el dispositivo que queremos conectar (en este caso un mouse, pero puede ser un celular, un teclado, etc.), en «modo visible» el asistente lo detectará y nos aparecerá listado con el código de su hardware. Nuevamente presionamos **Siguiente.**

4. La pantalla a continuación nos muestra el proceso de emparejamiento que habilitará al dispositivo en nuestro sistema.

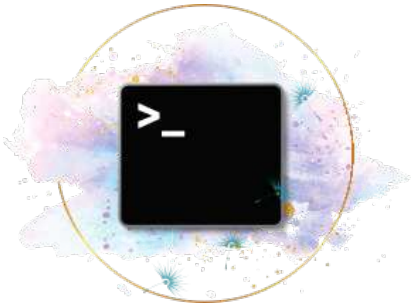
5. Una vez que se completa la conexión, el dispositivo queda conectado y añadido al listado de dispositivos del sistema. Finalmente, pulsamos **Cerrar.**

7.3 Soy programador/a ¿Cómo puedo participar en el desarrollo de Huayra?

Te invitamos a conocer todos nuestros desarrollos y a realizar aportes de código en https://github.com/HuayraLinux ¡animate a enviar un pull request!

Para conocer cómo es el procedimiento para crear un paquete .deb para Huayra, podés mirar el siguiente tutorial (nivel avanzado). Cada vez que sale una nueva versión de Huayra se actualizan, agregan y eliminan aplicaciones y paquetes de software. Todos los cambios entre versión y versión los encostrarás en nuestro changelog.

Capítulo 8 · La Terminal



8.1 “Dame una terminal y moveré el mundo.”

La Terminal de Huayra, también conocida como “consola”, es una herramienta para usuarios y usuarias más avanzados/as que permite un mayor control y velocidad a la hora de realizar ciertas tareas.

Existen al menos dos formas de abrir una terminal en Huayra. Para hacerlo en forma gráfica podemos ir al menú **Aplicaciones** -> **Herramientas del Sistema** -> **Terminal de Huayra**.

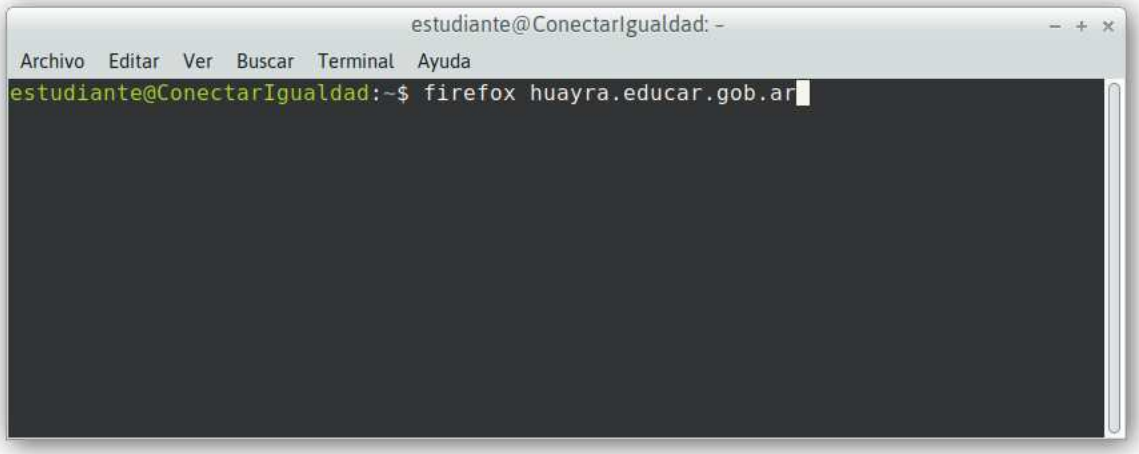
La otra forma es más rápido y se trata de usar un atajo de teclado, presionando simultáneamente las teclas: **Control** + **Alt** + **T** .

8.2 Comandos

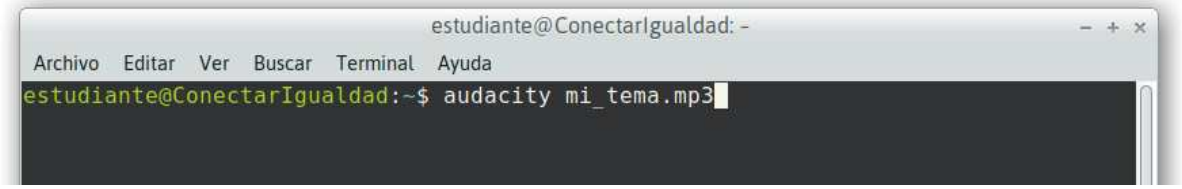
Para comenzar a entender la Terminal, vamos a ver algunos ejemplos, pero antes que nada, un truco útil: cuando estemos en la Terminal, podemos usar la tecla **TAB** para autocompletar comandos y nombres de archivo. Esto hace que su uso sea muchísimo más rápido y reducimos la posibilidad de equivocarnos al tipear.

Por ejemplo, si queremos lanzar la aplicación "**Firefox**":

- Abrimos una Terminal con **CONTROL + ALT + T**
- Escribimos "**fire**" y presionamos la tecla **TAB .**
- Mágicamente, se completó el comando y no es necesario terminar de escribir la palabra. Luego, y separado por un espacio, escribimos una URL de ejemplo.



Como vemos, al escribir el nombre de la aplicación, en este ejemplo el navegador Firefox, seguido de una URL logramos abrir el programa y la página, directamente. ¿Interesante, no? Descubrimos los parámetros, son detalles adicionales que le pasamos al comando para realizar la tarea, acá es una URL para navegar, otro ejemplo sería pasarle a Audacity el nombre de un archivo mp3 que queremos editar.



Desde la terminal podemos movernos por los distintos directorios o carpetas de nuestra máquina y hacer todo tipo de operaciones como copiar, mover y eliminar archivos y carpetas.

Para ello necesitamos conocer algunos **comandos**.

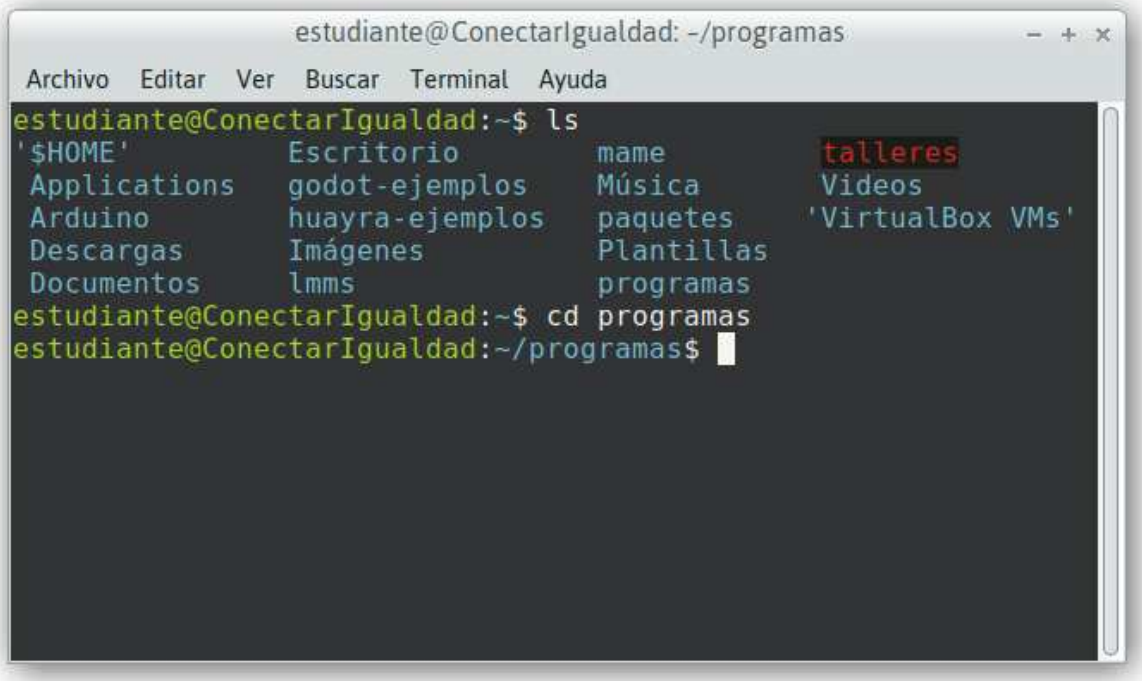
8.3 Comandos de archivos

Cuando abrimos la terminal, vamos a ver antes de prompr (**$**) este símbolo **~** . Significa que estamos “parados” en nuestra carpeta /home . En Linux, todos los archivos de un usuario están en /home/usuario (reemplazando “usuario” por el nombre del usuario). Para listar los archivos y carpetas que ahí se encuentran, vamos a usar el comando **ls**.

8.3.1 comando ls

8.3.2 comando cd

Para ingresar a un directorio o carpeta, usamos el comando “cd”, por ejemplo si quisiéramos ingresar a “programas”, el comando sería “cd programas” y presionamos **enter**. como vemos en la imagen:



Para volver al nivel anterior, usamos “cd ..”

8.4 resumen de comandos básicos

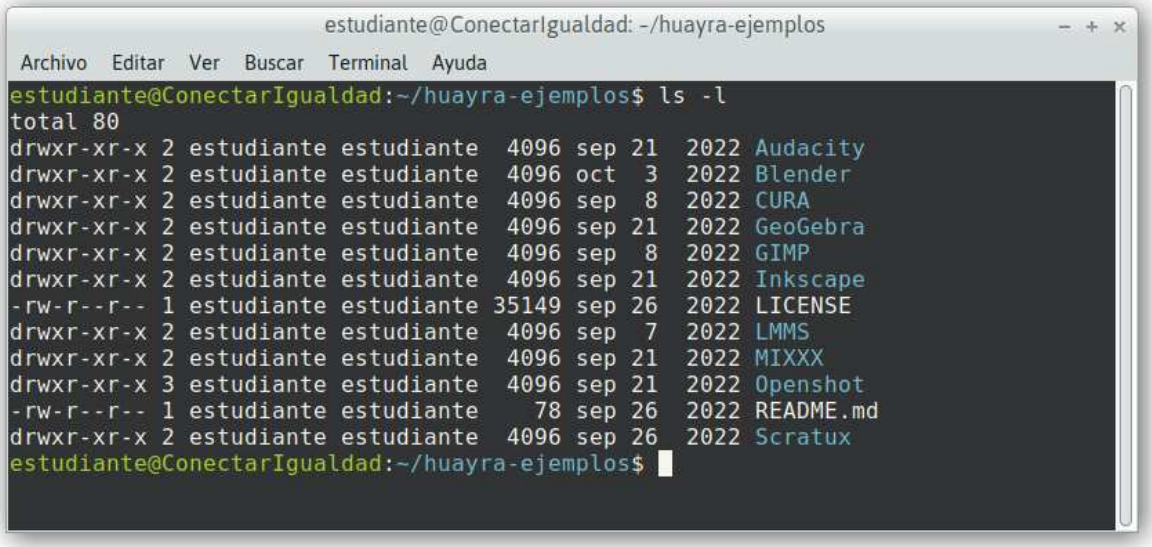
ls	Muestra el contenido del directorio actual.
cd	Cambia el directorio. Por ejemplo, cd Documentos te llevará al directorio "Documentos".
pwd	Muestra la ruta completa del directorio actual.
mkdir	Crea un nuevo directorio. Por ejemplo, mkdir dibujos creará un directorio llamado "dibujos".
touch	Crea un nuevo archivo. Por ejemplo, touch notas.txt creará un

	archivo llamado "notas.txt".
cp	Copia archivos y directorios. Por ejemplo, cp notas.txt backup/ copiará el archivo "notas.txt" al directorio "backup".
mv	Mueve archivos y directorios. Por ejemplo, mv texto.txt enero/ moverá el archivo "texto.txt" al directorio "enero".
rm	Elimina archivos y directorios. Por ejemplo, rm nota.txt eliminará el archivo "nota.txt". Cuidado con este comando, ya que los archivos eliminados no se pueden recuperar fácilmente.
cat	Muestra el contenido de un archivo en la pantalla. Por ejemplo, cat file.txt mostrará el contenido del archivo "file.txt".
nano	Editor de texto en la terminal. Podés usarlo para crear y editar archivos de texto.
man	Muestra el manual de ayuda para un comando específico. Por ejemplo, man ls mostrará la documentación del comando "ls".

Estos son solo algunos de los comandos básicos que te ayudarán a moverte y administrar archivos en la terminal de Huayra. A medida que te familiarices más con el sistema, podrás explorar otros comandos y sus opciones.

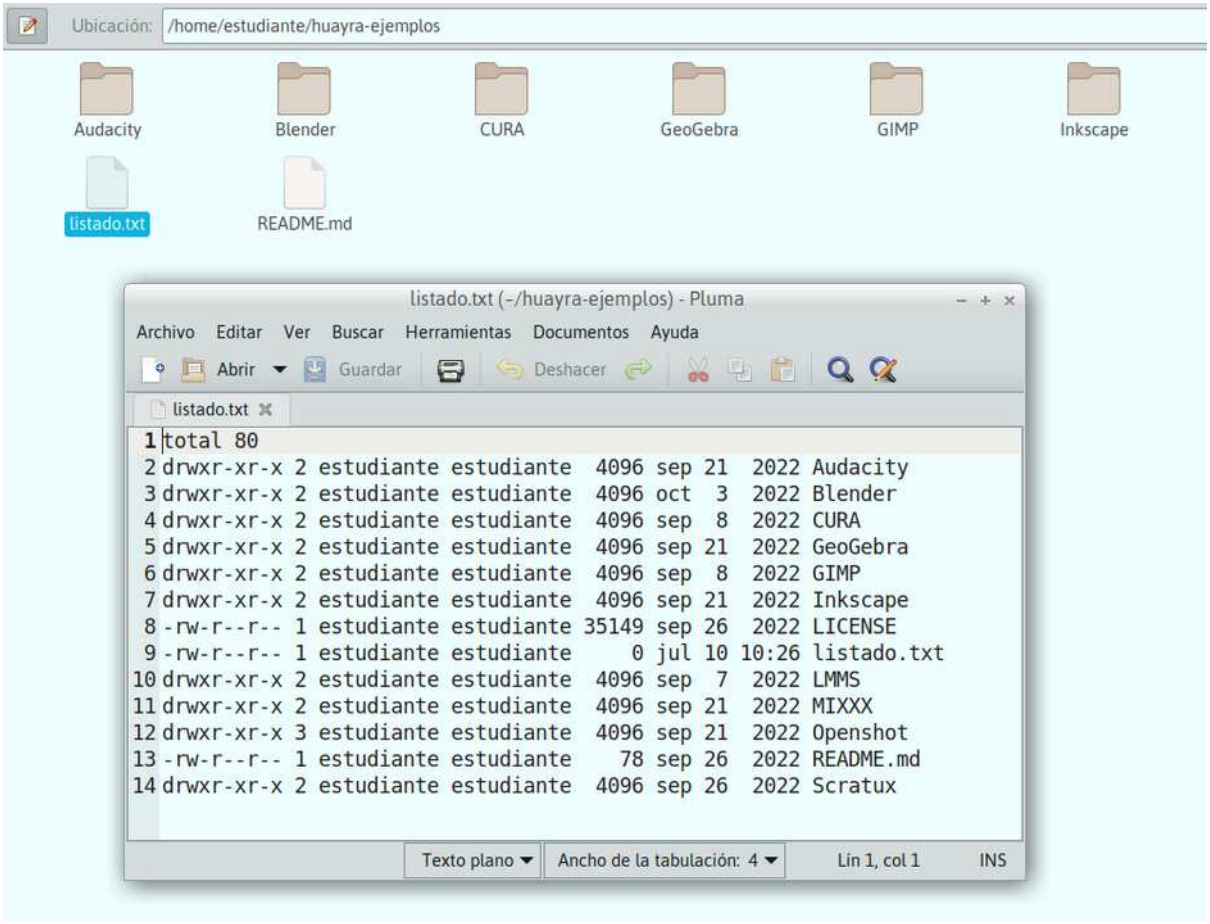
8.5 Enviar la salida de un comando a un archivo

Puede ser que necesitemos guardar el resultado de un comando, por ejemplo, en un archivo de texto plano. Podemos hacerlo direccionando la salida del mismo con "**>**", veamos un ejemplo:



Con el comando "ls -l" pedimos un listado del directorio, si quisieramos enviar el resultado a un archivo, hacemos: ls -l > archivo.txt

y listo, luego tenés el archivo "listado.txt" con el listado.



8.7 El comando grep

Este comando nos permite **buscar textos** dentro de un archivo o dentro de la salida estándar de otro comando. Veamos unos ejemplos:

$ grep hola datos.txt

Con esta instrucción buscamos todas las líneas que contengan la palabra “hola” dentro del archivo “datos.txt”. Este comando también puede filtrar la salida de otro comando, como veremos en un ejemplo a continuación.

8.7 El comando history

Con history podemos conocer el historial de comandos de la terminal, cada uno de los ingresos que hacemos quedan guardados en un archivo llamado ".bash_history" y podemos ir recorriéndolos (de último a primero) presionando la tecla "**flecha arriba**" en la terminal.

Si quisiéramos saber todos los comandos rm que fueron ejecutados con anterioridad, podemos hacer:

$ **history** | **grep** **rm**

Lo que usamos se llama "pipe" o "tubería", es el carácter barra vertical (**|**) y lo que hace es conectar la salida del comando **history** a la entrada del comando **grep**.

8.8 ¿Qué es un "script" ?

Un script Bash es un archivo de texto que contiene una serie de comandos escritos en el lenguaje de programación Bash. Bash (Bourne Again SHell) es un intérprete de comandos utilizado principalmente en sistemas operativos basados en Unix y Linux.

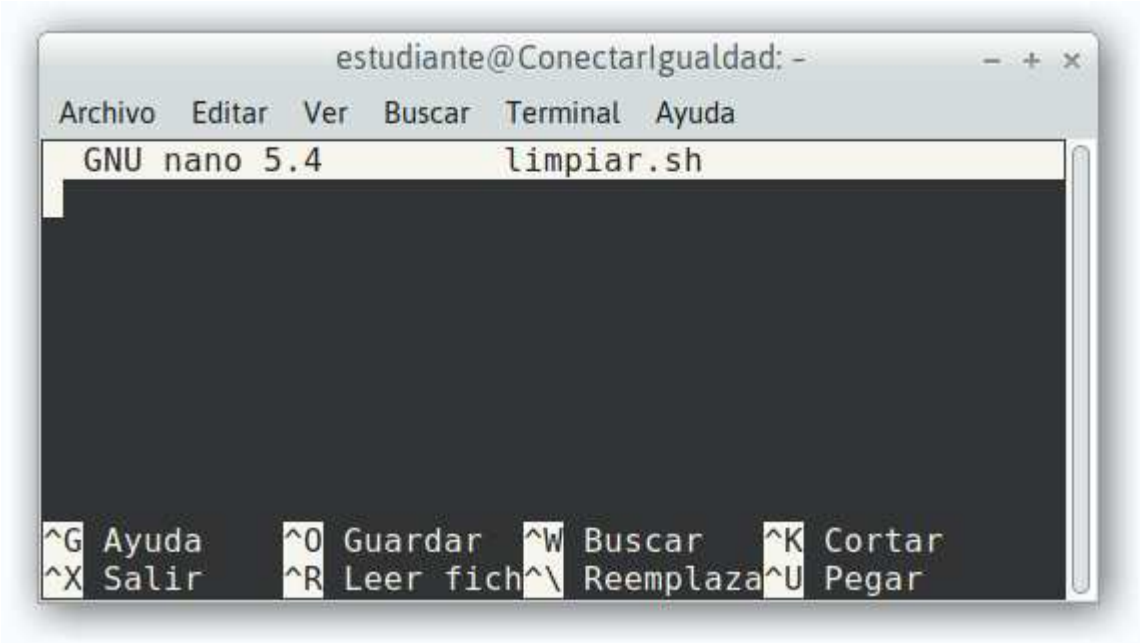
Los scripts Bash se utilizan para automatizar tareas y realizar secuencias de comandos en una terminal. Pueden contener variables, estructuras de control como bucles y condicionales, funciones y llamadas a otros programas o comandos del sistema.

Algunos ejemplos comunes de tareas que se pueden realizar con scripts incluyen la manipulación de archivos y directorios, el procesamiento de texto, la ejecución de comandos en secuencia y la automatización de tareas repetitivas.

Los scripts Bash son muy flexibles y poderosos, lo que los convierte en una herramienta útil para administradores de sistemas y usuarios avanzados que desean automatizar tareas y personalizar su entorno de línea de comandos.

A modo de ejemplo, vamos a hacer un script bash que libere espacio en disco. Para ello creamos un archivo de texto que vamos a llamar "limpiar.sh" con el siguiente comando:

$ nano limpiar.sh



Y escribimos dentro las siguientes líneas:

#!/bin/bash

echo "Eliminando paquetes instalados automáticamente que no son necesarios.”

sudo apt autoremove

echo "Borrando el caché de APT."

sudo apt clean

echo "Proceso de liberación de espacio FINALIZADO."

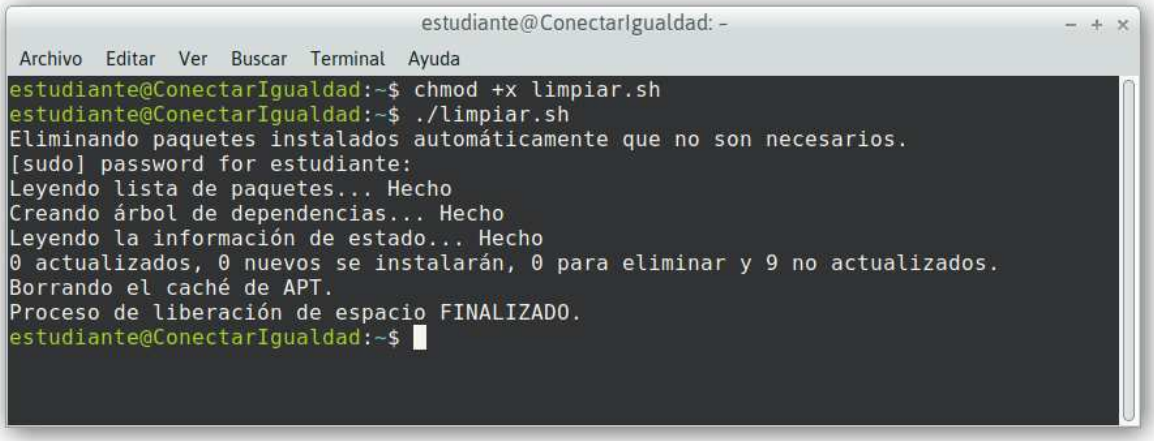
Grabamos el archivo en el editor NANO presionando **CONTROL** + "**O**" y luego **ENTER.**

Salimos del editor con **CONTROL** + "**X**"

Ya tenemos nuestro script listo, pero para poder ejecutarlo primero necesitamos darle permisos de ejecución. Para ello escribimos el comando:

$ chmod +x limpiar.sh

Ahora sí está todo listo. Ejecutamos el script escribiendo "**./limpiar.sh**" (pedirá la contraseña ya que estamos ejecutando los comandos como administrador).


estudiante@ConectarIgualdad: -
Archivo Editar Ver Buscar Terminal Ayuda
estudiante@ConectarIgualdad:~$ chmod +x limpiar.sh
estudiante@ConectarIgualdad:~$ ./limpiar.sh
Eliminando paquetes instalados automáticamente que no son necesarios.
[sudo] password for estudiante:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 9 no actualizados.
Borrando el caché de APT.
Proceso de liberación de espacio FINALIZADO.
estudiante@ConectarIgualdad:~$

Capítulo 9 · Configuración de Audio y video

9.1 Cómo conectar un monitor, televisor o proyector a nuestra netbook.

Si pensaste que al conectar la netbook a algún dispositivo no era compatible porque "**no hay señal**", puede ser que haya que cambiar algo en la configuración, veamos:

Lo primero que vamos a hacer es enchufar nuestra netbook con un cable HDMI al dispositivo que necesitamos utilizar.

Al hacerlo, Huayra toma ese dispositivo como una “segunda pantalla”, por lo que podemos posicionar ventanas en una u otra en forma independiente. Por ejemplo, si tomamos una ventana de la pantalla de la netbook y la arrastramos, comprobaremos que podemos pasarla a la nueva pantalla.

Sin embargo, muchas veces necesitamos que la segunda pantalla sea exactamente la misma que la de la netbook, por ejemplo al utilizar un proyector.

A continuación vamos a ver cómo configurar Huayra para que se vea lo mismo en ambas pantallas en forma simultánea.

Configurar pantalla

Para comenzar vamos a **Sistema > Preferencias > Hardware > Pantallas,** se nos abrirá la siguiente ventana:

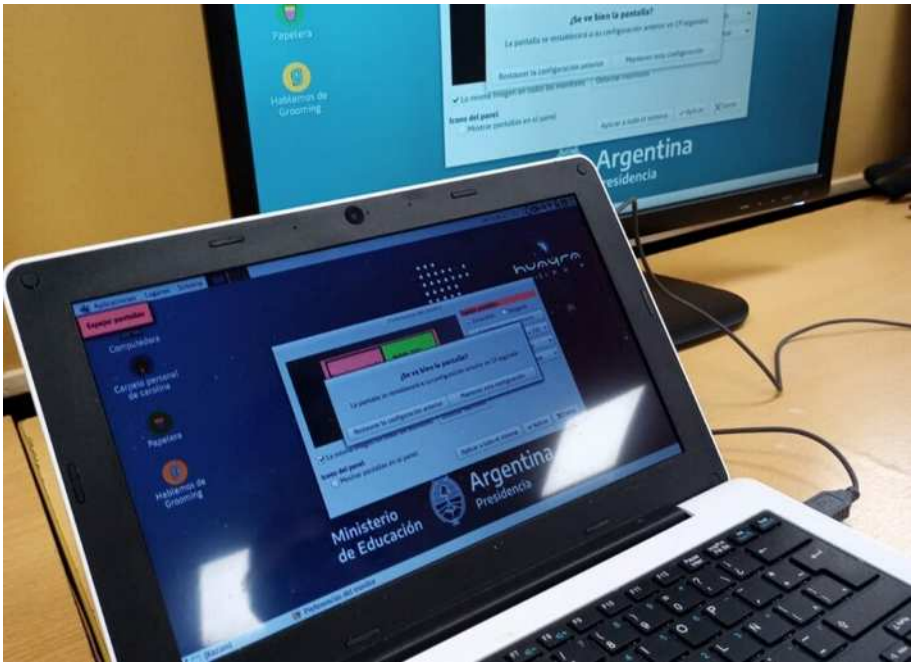
Portátil es el nombre de la pantalla de nuestra netbook. Y vemos otro dispositivo, en este caso un monitor BGO, pero que no está seleccionado.

El paso siguiente es **seleccionar el dispositivo** que conectamos y cliquear en **Encendido** y luego **Aplicar.**

Ahí aparecerá la pantalla de nuestra netbook replicada en el dispositivo. Y una ventana donde nos da la opción de **restaurar la configuración anterior** o **mantener esta configuración,** damos click en "**mantener esta configuración**".

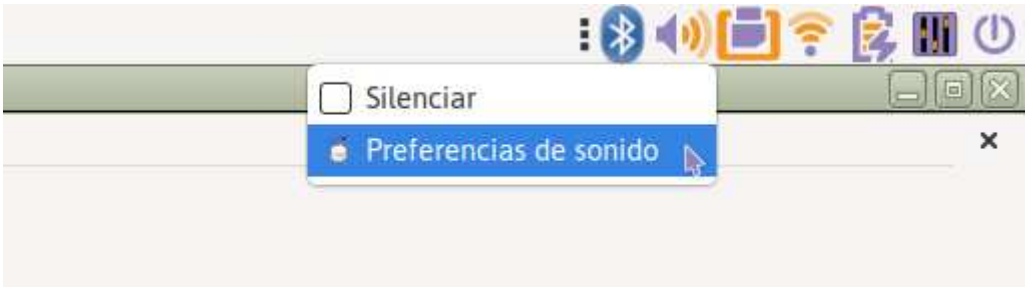
También tenemos la opción de **espejar pantallas.** Esta opción nos permite ver directamente reflejadas las acciones realizadas en la *netbook* en el dispositivo. Para ello clickeamos en "**La misma imagen en todos los monitores."**

¡Y listo, ya tenemos nuestra pantalla configurada!

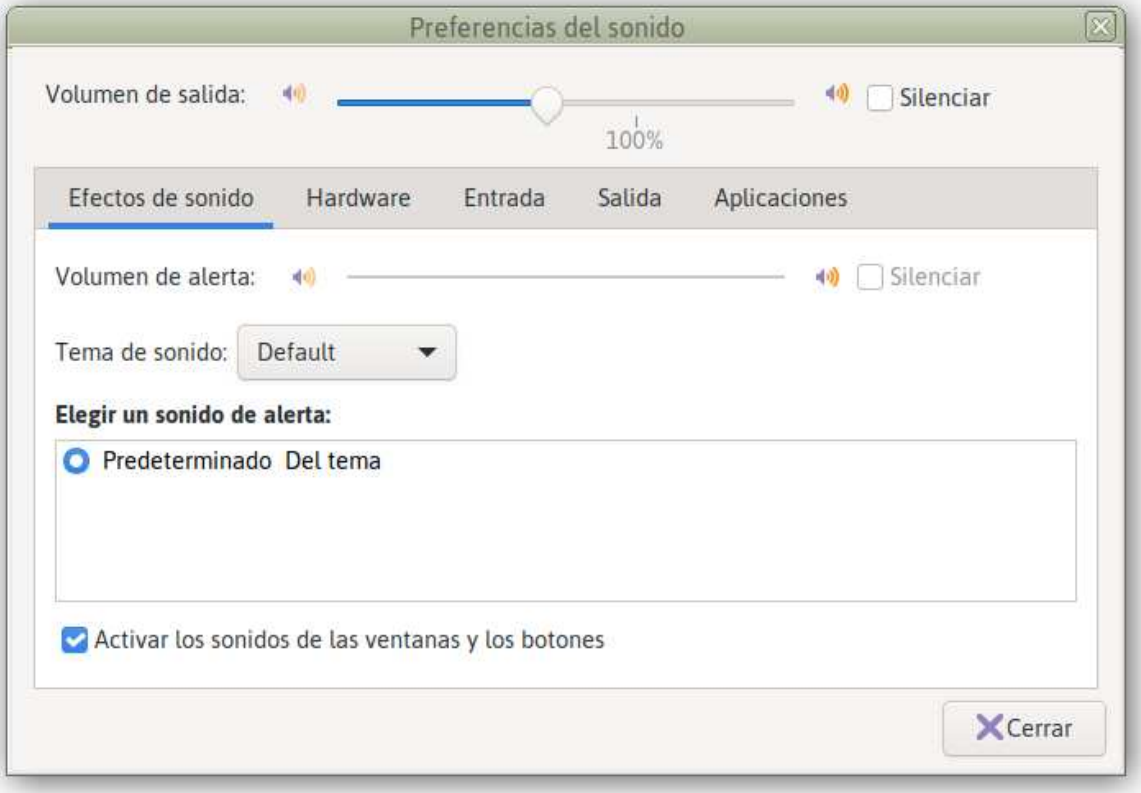


9.2 Configuración de salida de Audio

Para acceder a la pantalla de configuración de dispositivos de audio, buscamos el icono de Parlante en la esquina superior derecha de la pantalla, (el mismo desde donde subimos y bajamos el volumen) y hacemos **clic derecho** → **Preferencias de Sonido**.

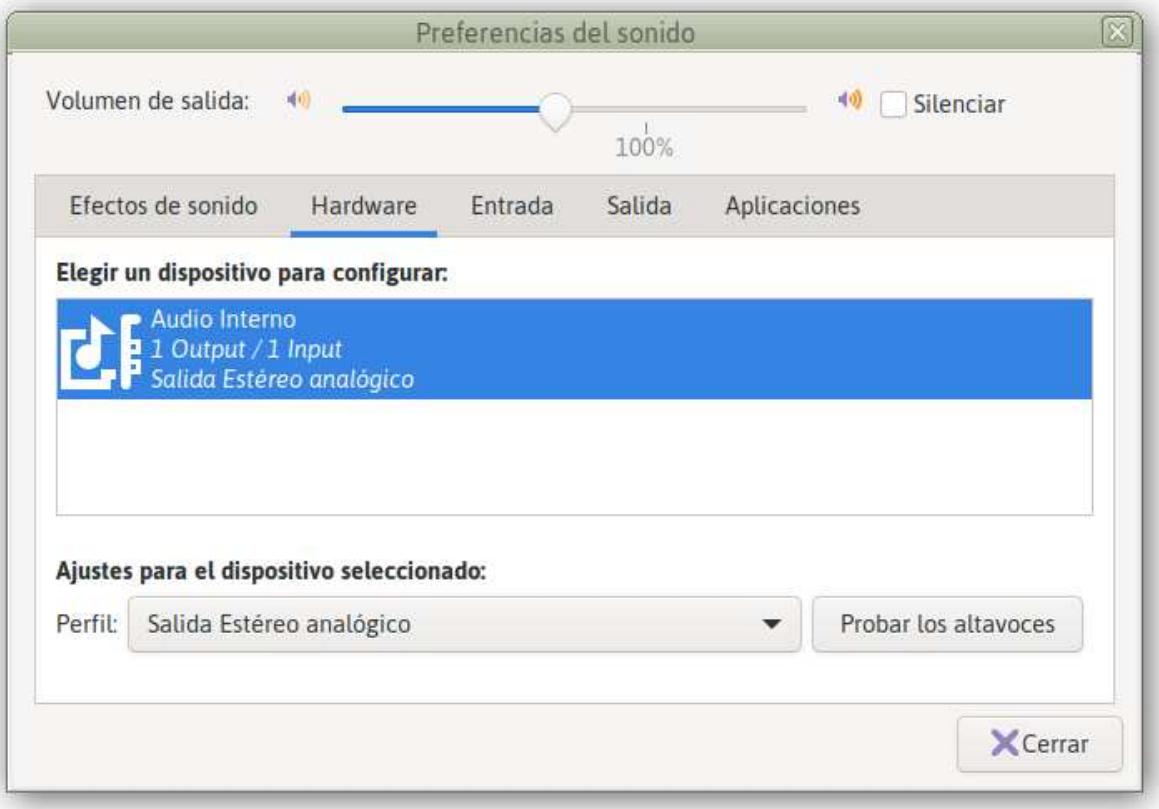


Nos abrirá la siguiente ventana:

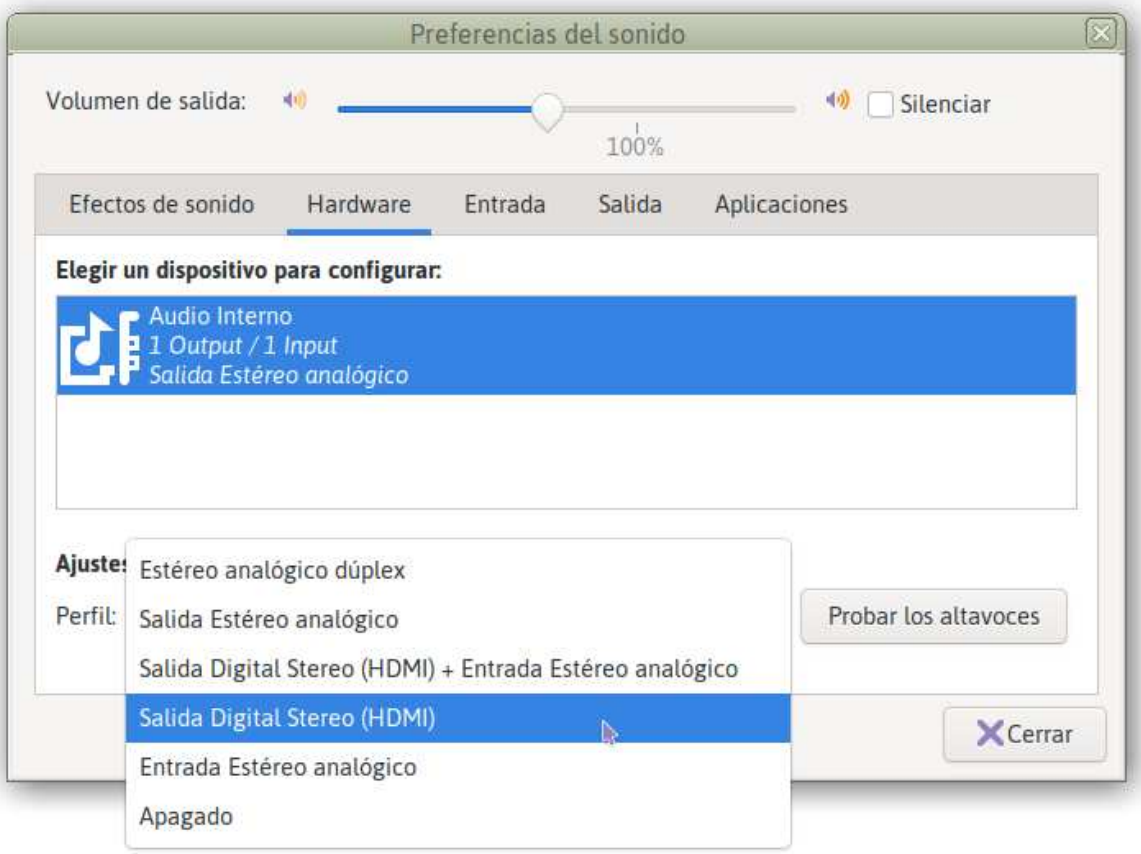


Dicha ventana tiene una serie de solapas, en la primera “Efectos de

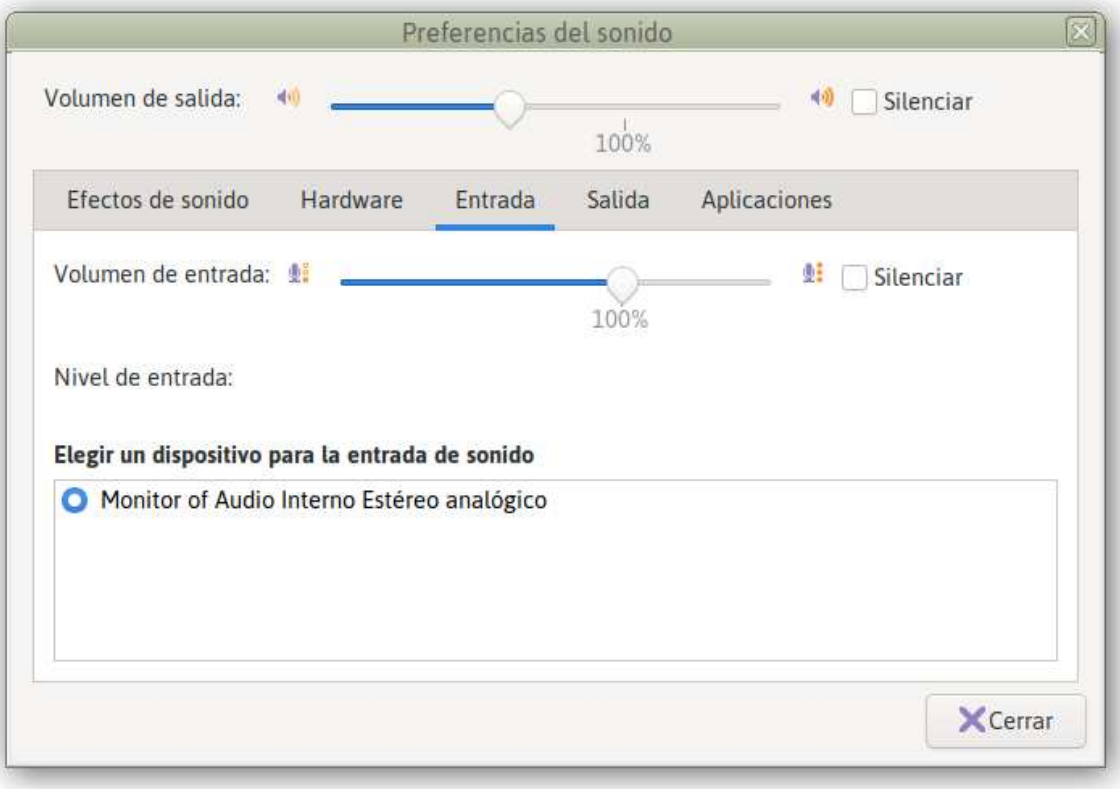
sonido” podremos configurar aspectos relacionados a los sonidos y alertas propios del sistema operativo. Pasemos a la siguiente solapa, “Hardware”.



Acá podemos seleccionar qué placa de sonido utilizar en el caso de que tuviéramos más de una, por ejemplo si conectáramos una placa de sonido externa vía USB. Luego, los dispositivos suelen tener más de una salida, por ejemplo la placa interna de sonido de las nets tienen la “salida Estéreo analógico” y también “Salida digital estéreo (HDMI)” , entonces si conectamos la netbook a un televisor con un cable HDMI y queremos que el sonido salga por la TV, tendremos que seleccionar “**Salida digital estéreo (HDMI)**” como salida.



Si lo que queremos es ajustar el **volumen del micrófono**, tenemos que ir a la pestaña "**Entrada**".

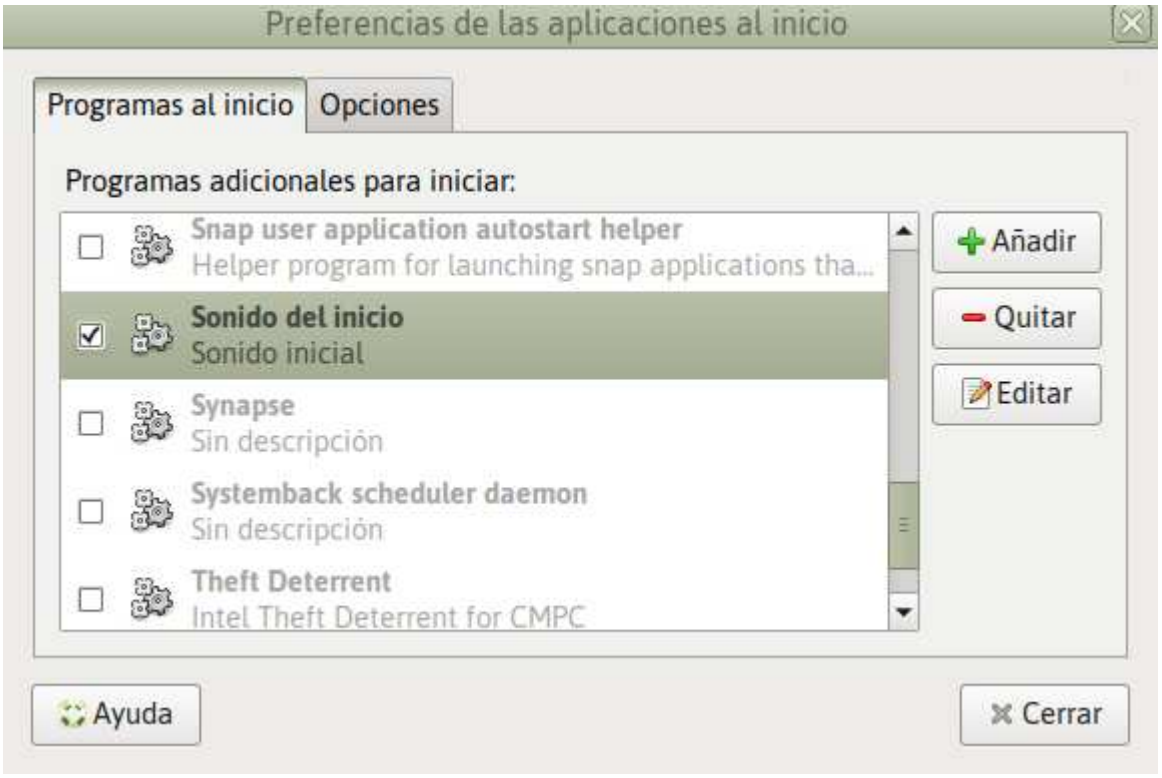


9.3 Desactivar el sonido de inicio de Huayra

Por distintos motivos puede ser que necesitemos que Huayra no haga su sonido característico de inicio, por ejemplo si solemos utilizar la compu de madrugada y el resto de la familia duerme. Vamos a contarte cómo hacer para desactivarlo en forma permanente.

Vamos al menú **Sistema** -> **Preferencias** -> **Personal** -> **Aplicaciones al inicio**

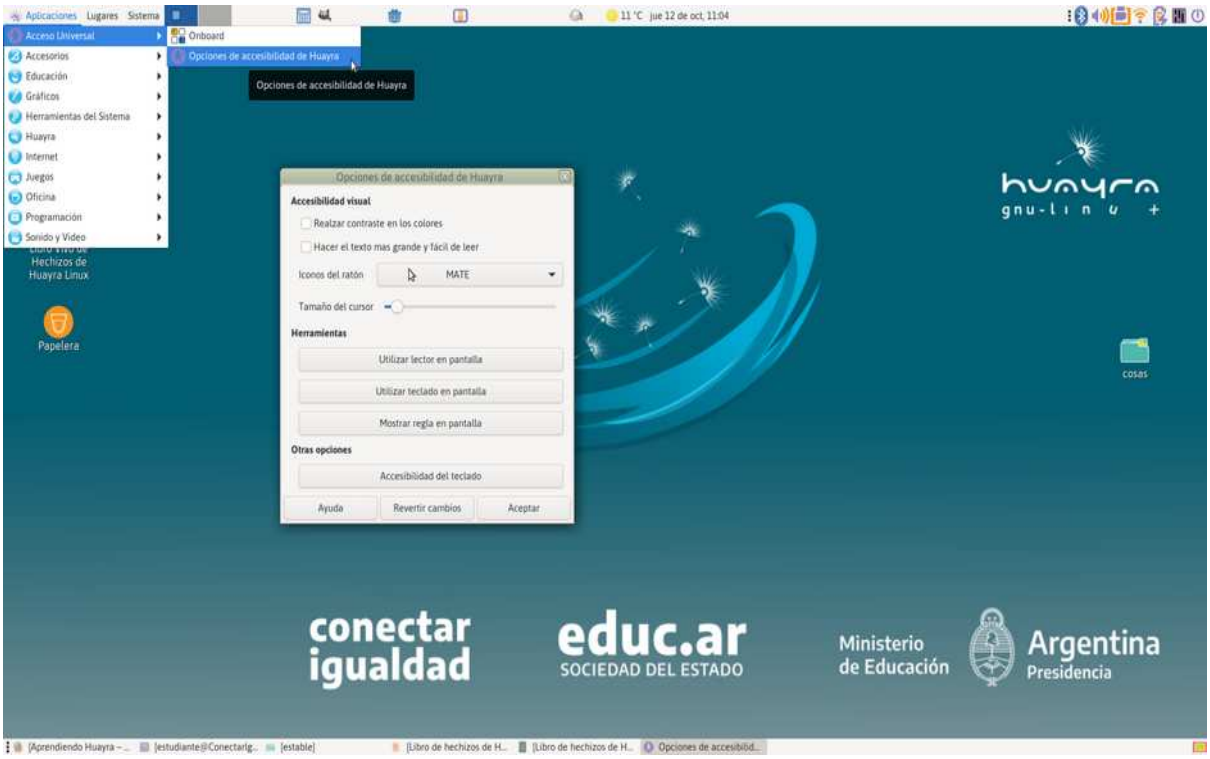
Y buscamos la opción "**Sonido del inicio**". La destildamos y presionamos "**Cerrar**".



Listo. Cuando iniciemos nuevamente Huayra ya no escucharemos el sonido.

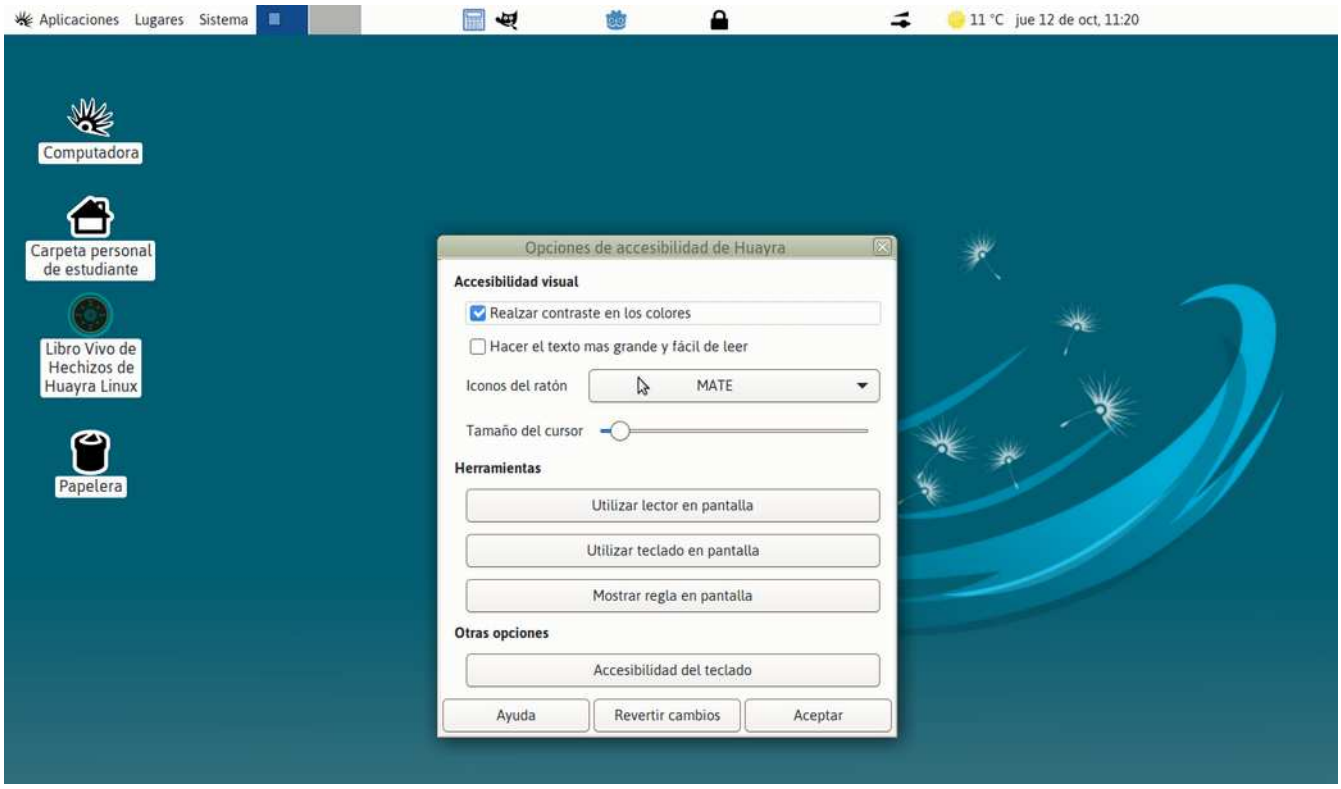
Capítulo 10 · Accesibilidad

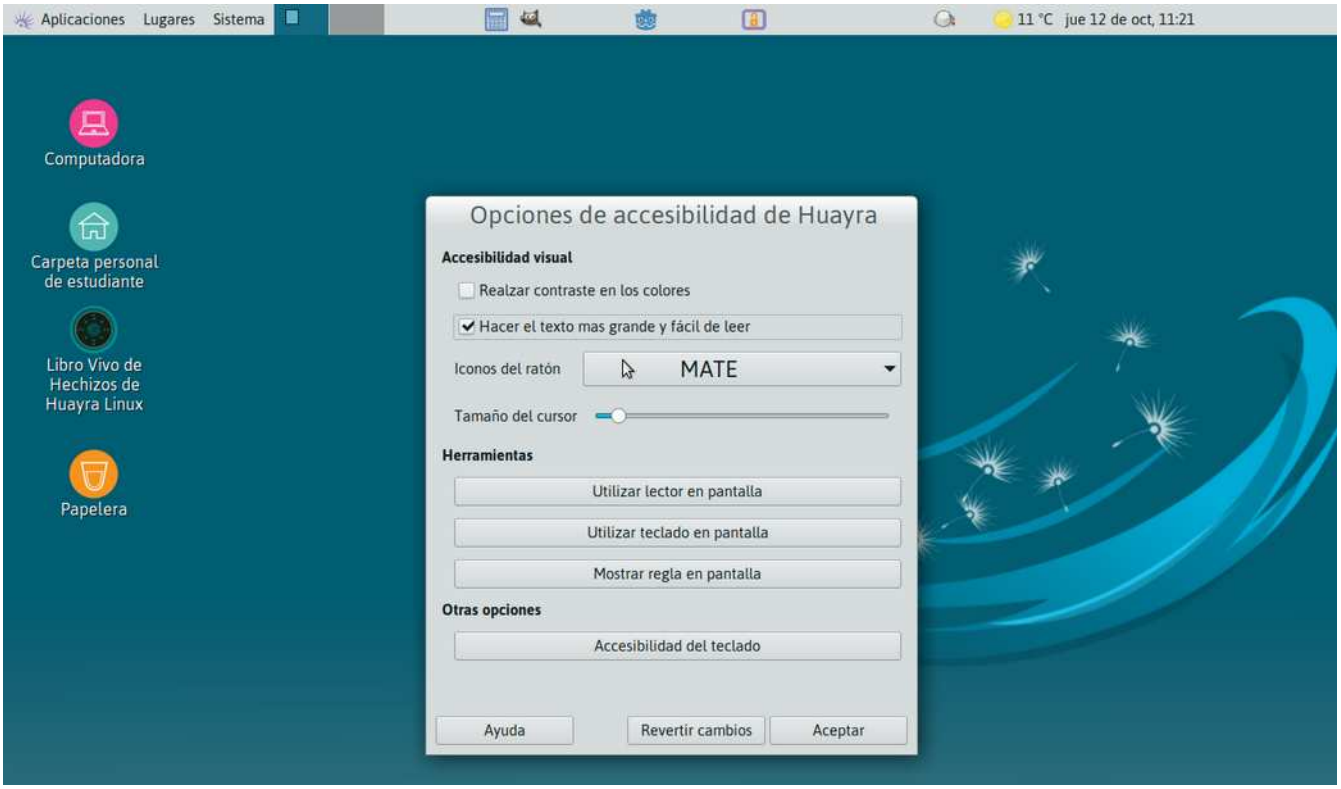
Para utilizar las funciones de accesibilidad de Huayra, nos dirigimos al menú **Aplicaciones** > **Acceso universal** > **Opciones de accesibilidad de Huayra** y desde este menú podemos habilitar o deshabilitar cada una de las opciones.

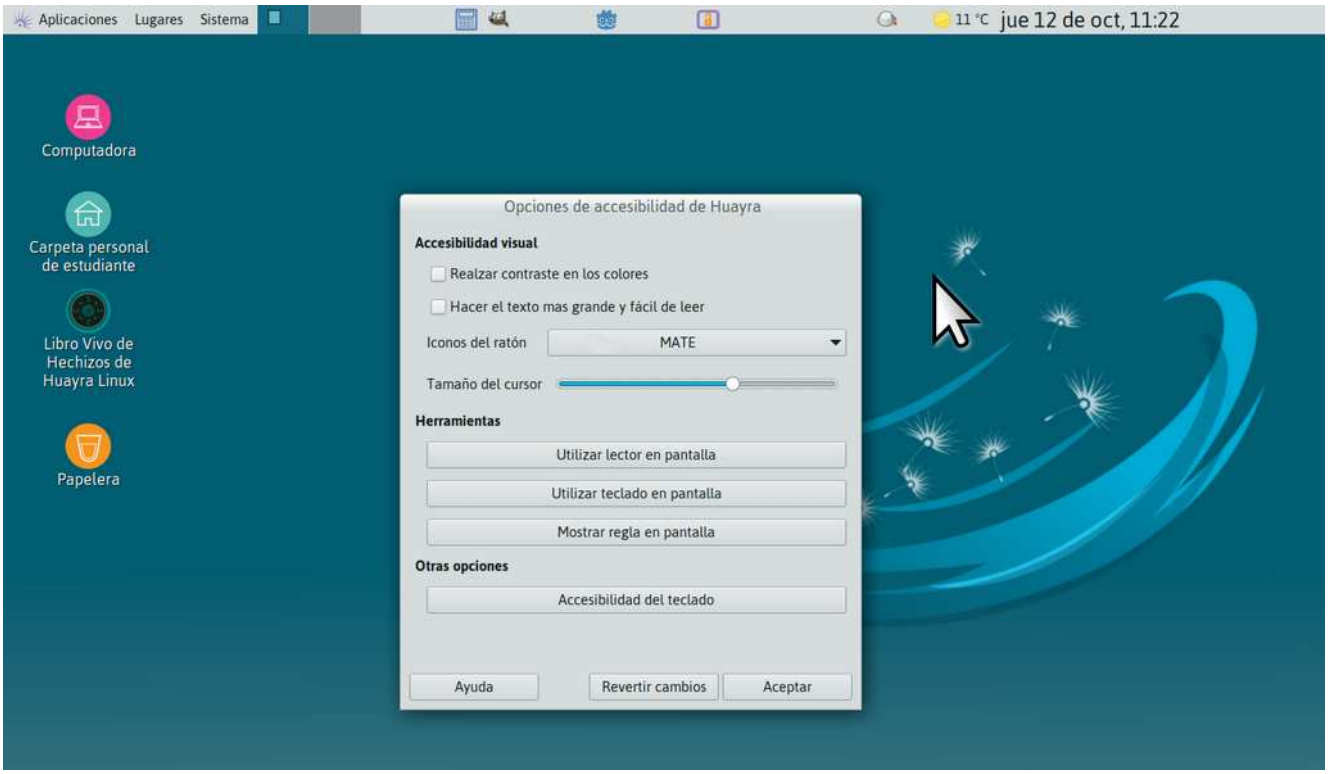


10.1 Contraste de colores, agrandar texto y modificar puntero

Podemos modificar el aspecto visual de Huayra para que se más accesible, por ejemplo, realzar el contraste de los colores, agrandar el texto para que sea más fácil de leer, modificar el tamaño del puntero del ratón para poder encontrarlo mejor.

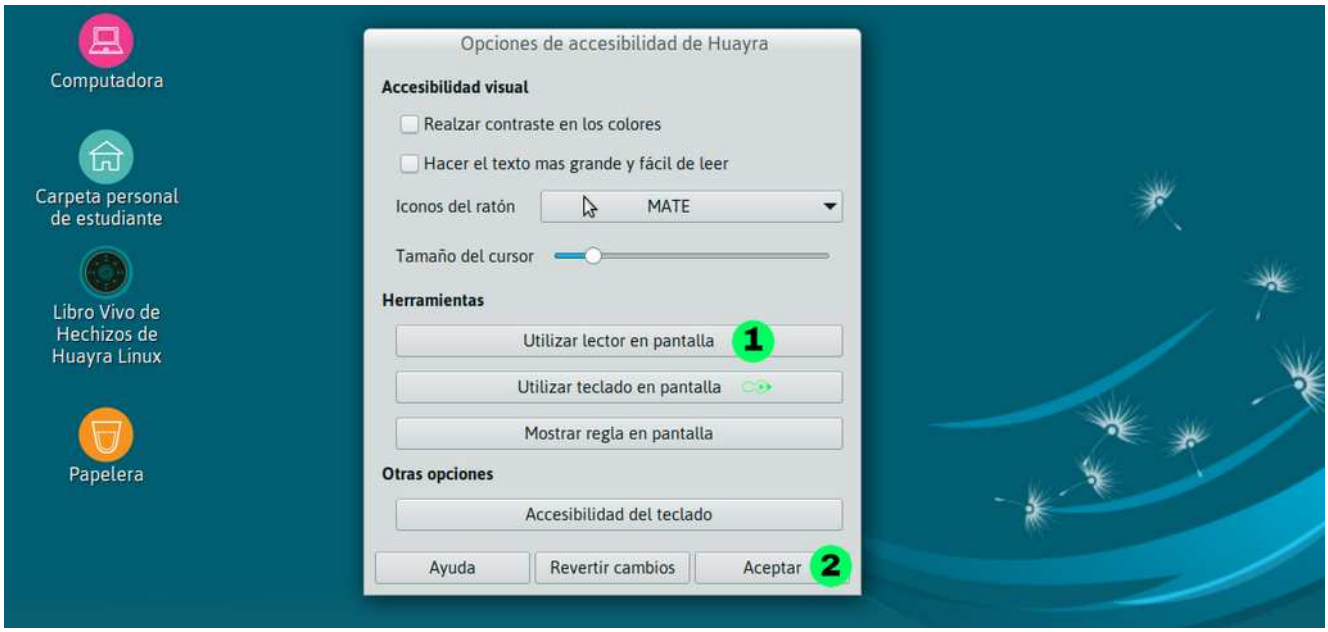






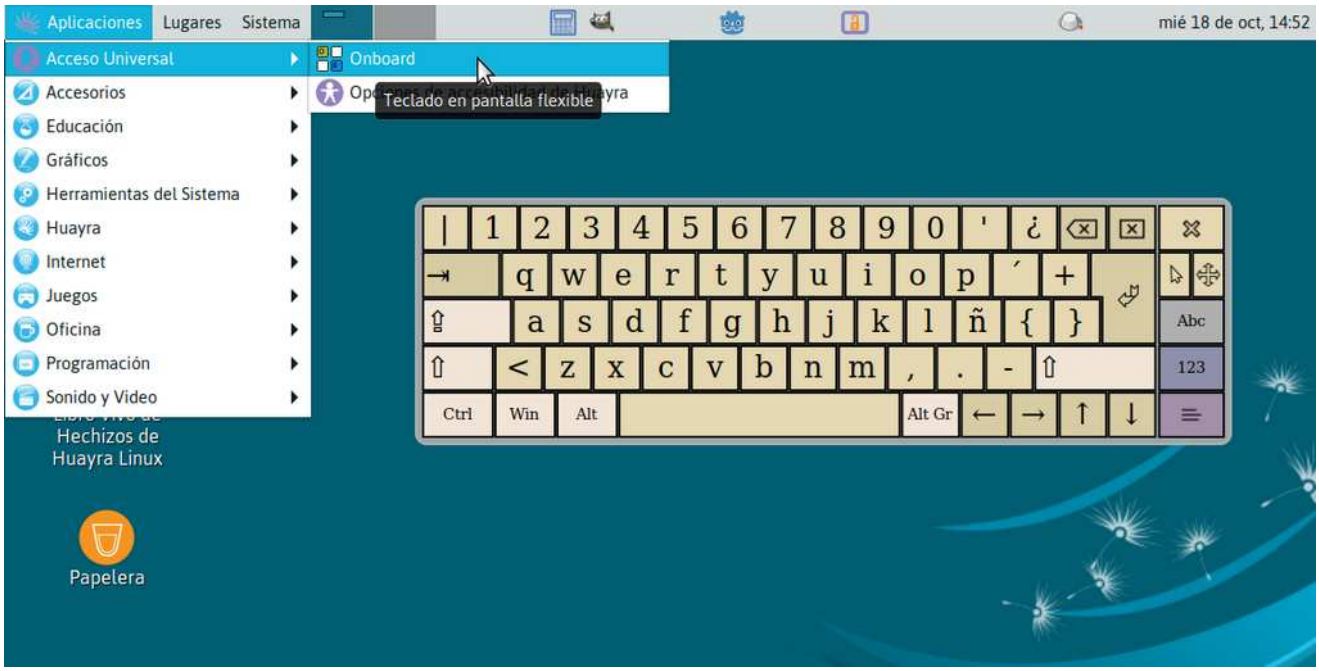
10.2 Lector de pantalla

El Lector de Pantalla es una aplicación que se encarga de leer en voz alta los textos activos de la pantalla. Para activar esta función tildamos la opción "**Utilizar lector en pantalla**" y presionamos "**Aceptar**", una vez hecho esto debemos **reiniciar la sesión**. Una vez que reinicie el lector de pantalla estará activo.



10.3 Teclado en pantalla

Otra opción a utilizar es el teclado en pantalla, esto nos permite simular la escritura de un teclado, pero usando el ratón. Para activarla, primero activamos la opción "**Utilizar teclado en pantalla**" y luego ejecutamos la aplicación "**ONBOARD**".



Apéndice A
Obsolescencia programada

¿Qué es la Obsolescencia Programada?

La obsolescencia programada es una estrategia industrial y comercial en la cual los productos son diseñados deliberadamente con una vida útil limitada para que se vuelvan obsoletos o inutilizables después de un período específico. Esta estrategia obliga a los consumidores a comprar nuevos productos más frecuentemente, generando un ciclo continuo de consumo. Veamos concretamente algunos casos reales.

En la década de 1920, el cartel Phoebus, del cual General Electric era miembro, acordó limitar la vida útil de las lamparitas o bombillas a 1,000 horas. Este fue uno de los primeros ejemplos documentados de obsolescencia programada. Las bombillas producidas antes de este acuerdo podían durar hasta 2,500 horas o más.

En 2017 una muy conocida compañía de teléfonos y computadoras de alta gama admitió que enlentecía sus celulares más antiguos a través de actualizaciones de software. La empresa alegó que esto era para prevenir apagones inesperados debido a baterías envejecidas, aunque muchos lo vieron como una táctica para impulsar la venta de nuevos dispositivos.

Una muy conocida empresa que vende impresoras ha sido acusada de implementar chips en sus cartuchos de tinta que indican que están vacíos cuando aún contienen tinta. Esta práctica obliga a los usuarios a comprar cartuchos nuevos más frecuentemente de lo necesario, incrementando las ventas de consumibles. Su competencia incluye chips en sus impresoras que determinan un "final de vida útil" después de un número predeterminado de impresiones, independientemente del estado real de la impresora. Una más sobre impresoras, otra estrategias de obsolescencia programada es que algunos fabricantes diseñan sus impresoras para que no funcionen si cualquiera de los cartuchos está vacío, entonces sucede que queremos imprimir en negro y la impresora no nos deja porque está vacío el color CYAN, por ejemplo. Otro modo de forzar la compra de consumibles.

La obsolescencia programada es una práctica controvertida que beneficia a las empresas al incrementar las ventas, pero que también genera desperdicio y frustración entre los consumidores.

“Cualquier tecnología suficientemente avanzada es indistinguible de la magia”.

- **Arthur C. Clarke**

“No es magia. Es ciencia, federalismo, soberanía, inclusión
y mucha pasión”.

- Equipo Huayra Linux